Software User Manual

57712-k/578xxS

Third-party information brought to you courtesy of Dell®

Switch Independent Partitioning Setup Guide

2CS57712-SWUM103-R

5300 California Avenue • Irvine, CA 92617 • Phone: 949-926-5000 • Fax: 949-926-5203

June 07, 2012

Revision History

Revision	*Date*	*Change Description*
2CS57712-SWUM103-R	06/07/12	**Updated:** • T7.2 software release by adding the 578xxS device family and VMWare ESXi 5.0 information.
2CS57712-SWUM102-R	12/08/11	**Updated:** • T6.4 software release by adding FCoE, DCB, VMWare® ESX/ ESXi 4.1, and BACS4 information.
2CS57712-SWUM101-R	3/31/11	**Updated:** • "Switch Independent Partitioning" in title and footer • "NIC Partition", "NIC Partitioned", and "NIC Partitioning" to "Switch Independent Partitioning" throughout the document • "Single Function" to "SF" throughout the document
2CS57712-SWUM100-R	03/15/11	Initial release

Broadcom Corporation
5300 California Avenue
Irvine, CA 92617

Table of Contents

About This Document......6
Purpose......6
Audience......12
Acronyms and Abbreviations......13
Technical Support......13
Configuring Switch Independent Partitioning......15
Using the Unified Server Configurator......15
Supported Operating Systems......26
Supported Server Platforms......27
Viewing and Configuring the Partitions......28
Windows Server 2008 R2......28
Installing the Latest Dell Drivers......28
Viewing the Enabled Devices in Device Manager......31
Broadcom Advanced Control Suite 4 (BACS4)......35
Microsoft Windows Network Connections......43
Device PCIe Bus Location......46
Red Hat Enterprise Linux v6.x......52
VMWare ESX/ESXi 4.1......57
Details on Configuring iSCSI Offload on ESX/ESXi 4.1......61
VMWare ESXi 5.0......67
Details on Configuring iSCSI Offload on ESXi 5.0......68
Details on Configuring FCoE Offload on ESXi 5.0......72
Details on Viewing Networking and Storage Performance on ESX......78
Setting MTU Sizes......82
Setting MTU Sizes in Windows......82
Setting MTU Sizes in Linux......85
Setting MTU Sizes in VMWare ESX/ESXi 4.1......89
Setting MTU Sizes in VMWare ESXi 5.0......92
Examples......94
Equal Oversubscription Example......95
Partitioned Oversubscription Example......102
Weighted Oversubscription Example......105
Oversubscription With One High Priority Partition Example......107
Default Fixed Subscription Example......109

Mixed Fixed Subscription and Oversubscription Example 111
Mixed Weights and Subscriptions Example 115

List of Tables

Table 1: Protocols Available in Operating Systems Versus SF and Switch Independent Partitioning Modes .. 26
Table 2: Dell 12G Networking Device Release verses Dell Servers 27
Table 3: Port, Function, MAC Address Example 47
Table 4: Non-DCB Equal Oversubscription 95
Table 5: DCB Equal Oversubscription 98
Table 6: DCB Equal Oversubscription with one Lossless FCoE Offload 100
Table 7: Non-DCB Partitioned Oversubscription 102
Table 8: Non-DCB Weighted Oversubscription 105
Table 9: Non-DCB Oversubscription With One High Priority Partition 107
Table 10: Non-DCB Default Fixed Subscription 109
Table 11: Non-DCB Mixed Fixed Subscription and Oversubscription 111
Table 12: DCB Mixed Fixed Subscription and Oversubscription with Lossless FCoE Offload 113
Table 13: Non-DCB Mixed Fixed Subscription and Oversubscription 115

About This Document

Purpose

This document provides instructions on how to enable Switch Independent Partitioning (also referred to as NPAR in other documentation) on Dell® PowerEdge® Servers installed with the Broadcom® 57712-k Blade Converged Network Daughter Card (NDC) Dual Port 10 GbE A Fabric option (see Figure 1) and the 578xxS device family (see Figure 2: "57800S Rack Converged NDC," on page 7, Figure 3: "57810S-k Blade Converged NDC," on page 8, Figure 4: "57810S-k Blade Converged Mezzanine," on page 9. Figure 5: "57810S Converged NIC," on page 10), Figure 6: "57810S-T 10GBASE-T Converged NIC," on page 11 and Figure 7: "57800S-T 10GBASE-T Rack Converged NDC," on page 12.

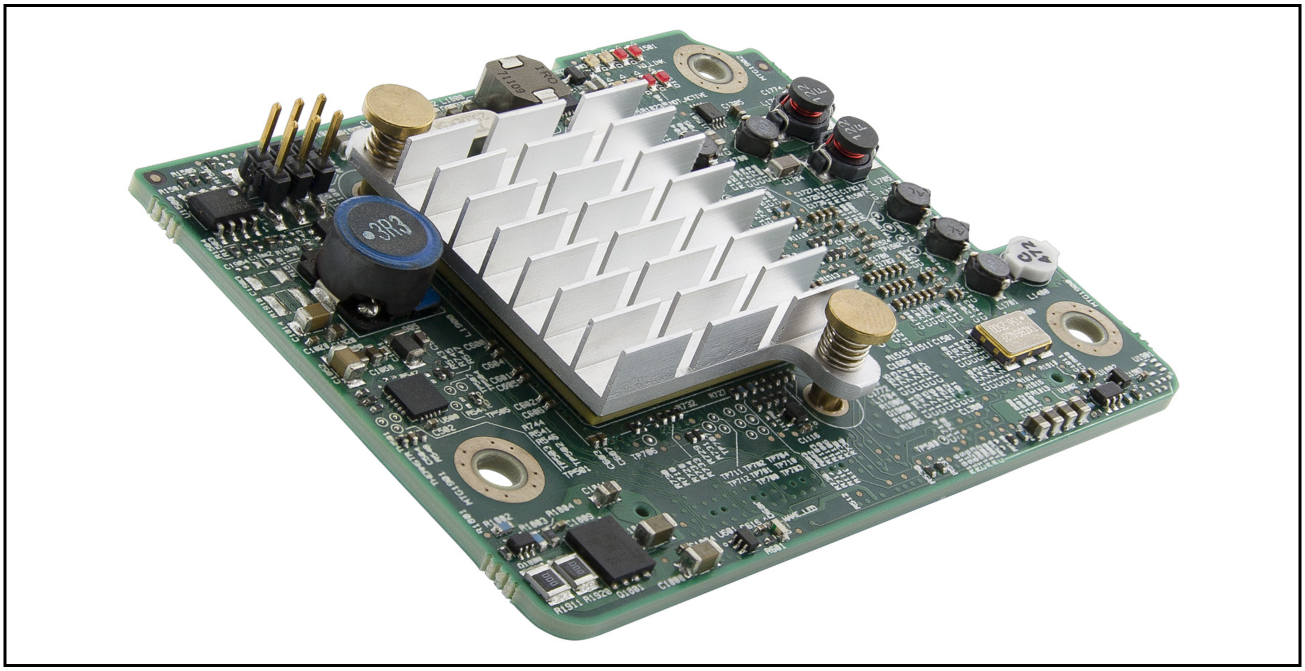


Figure 1: Broadcom 57712-k Blade Converged NDC

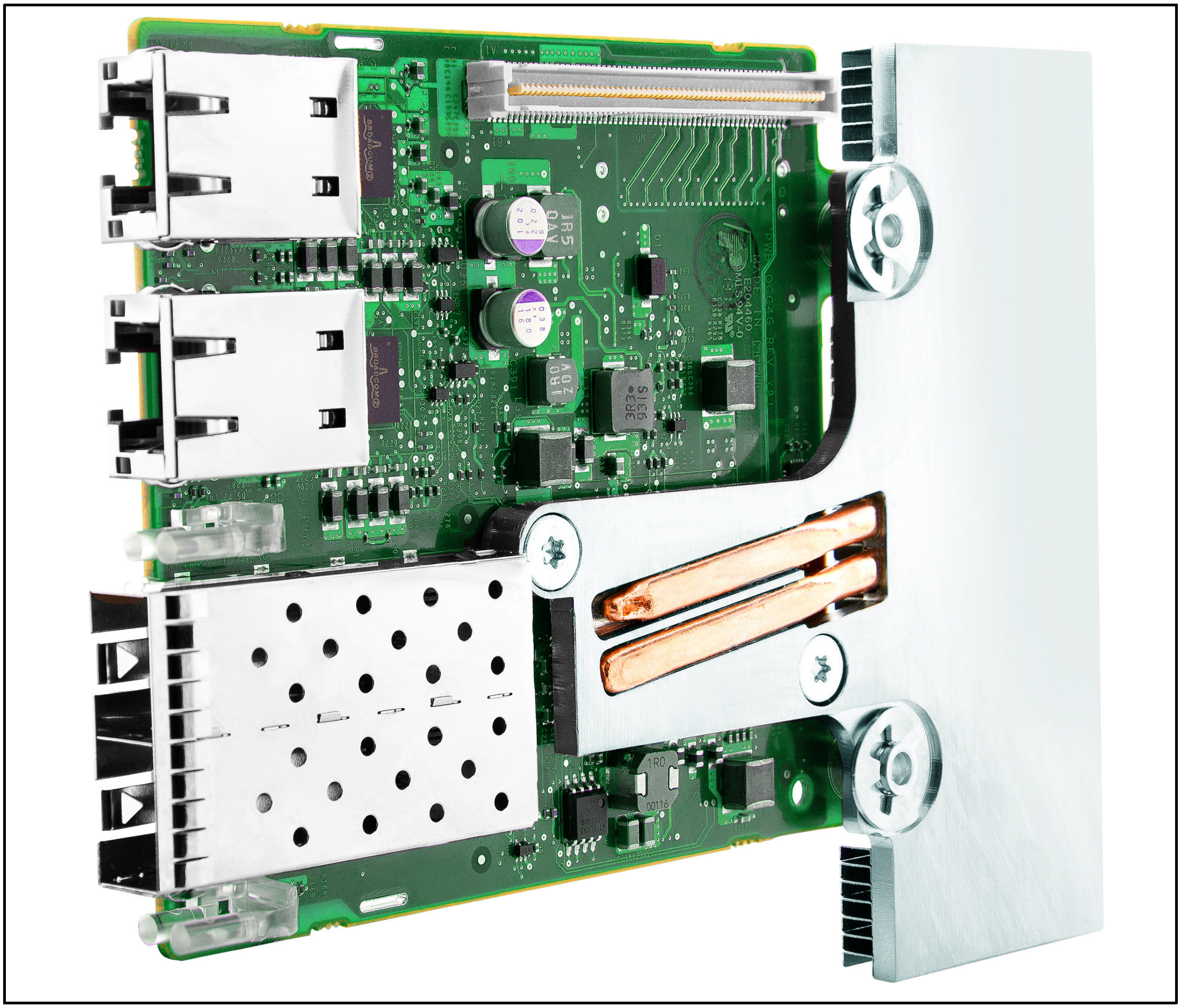


Figure 2: 57800S Rack Converged NDC

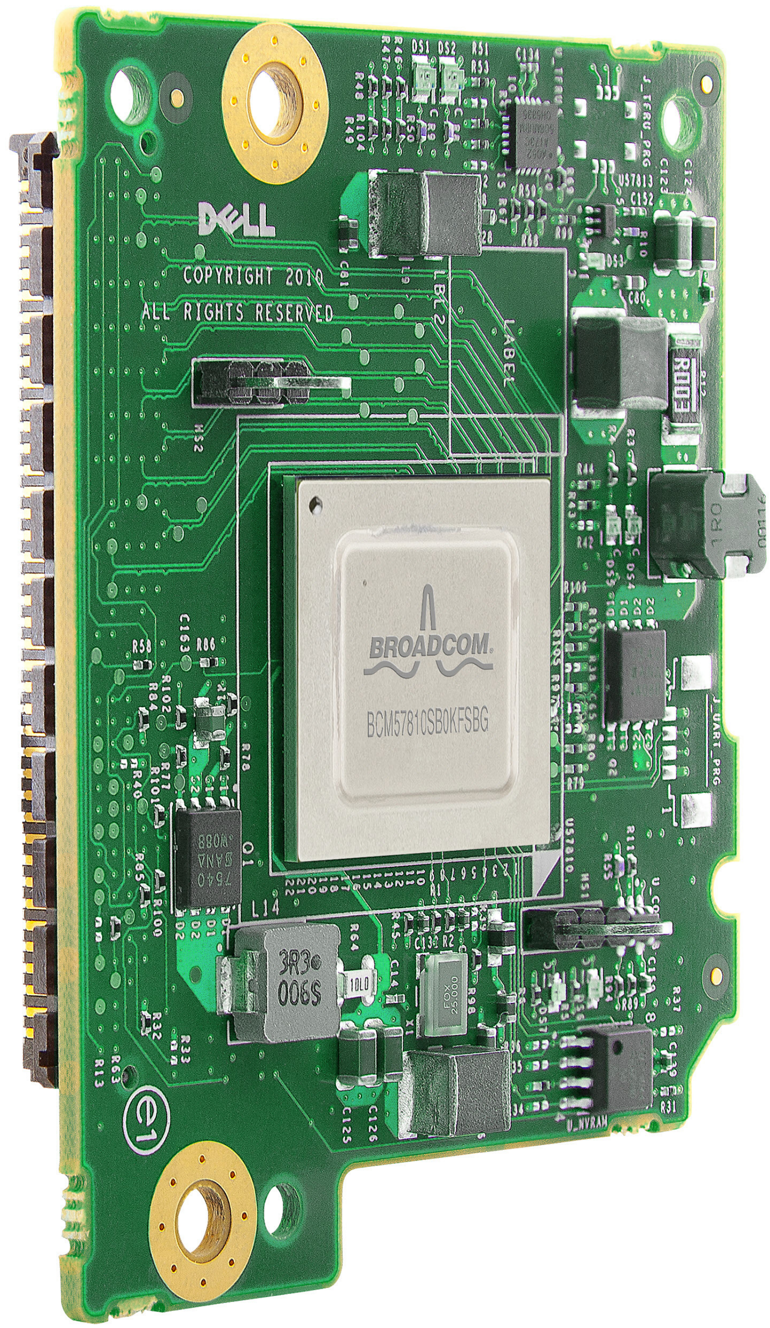


Figure 3: 57810S-k Blade Converged NDC

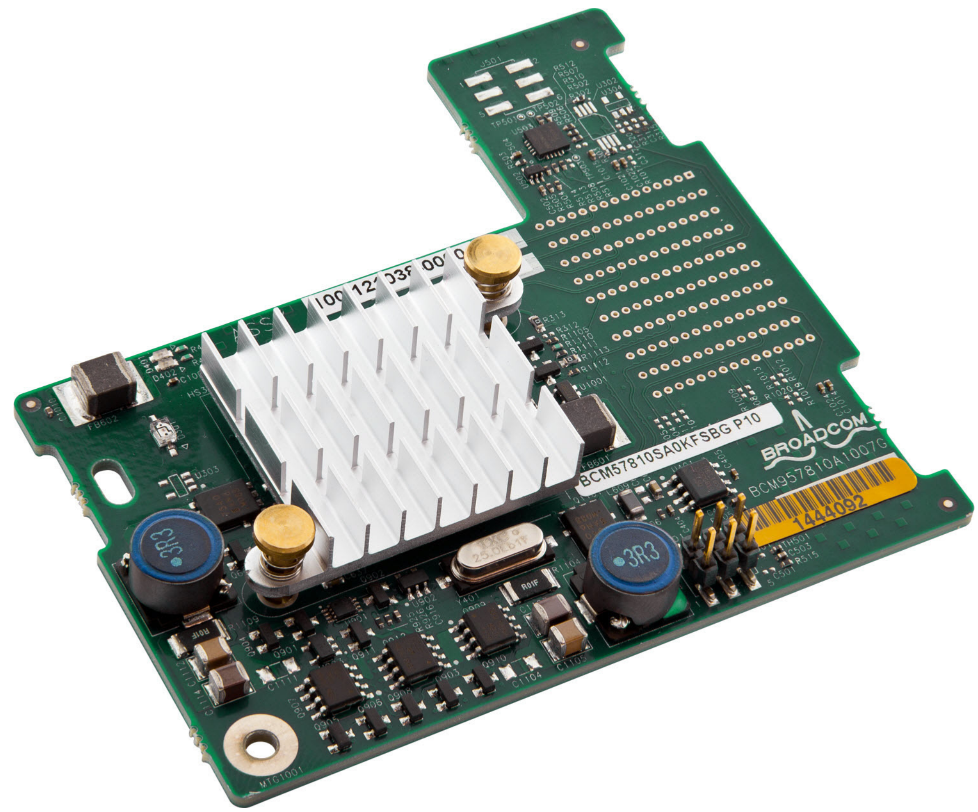


Figure 4: 57810S-k Blade Converged Mezzanine

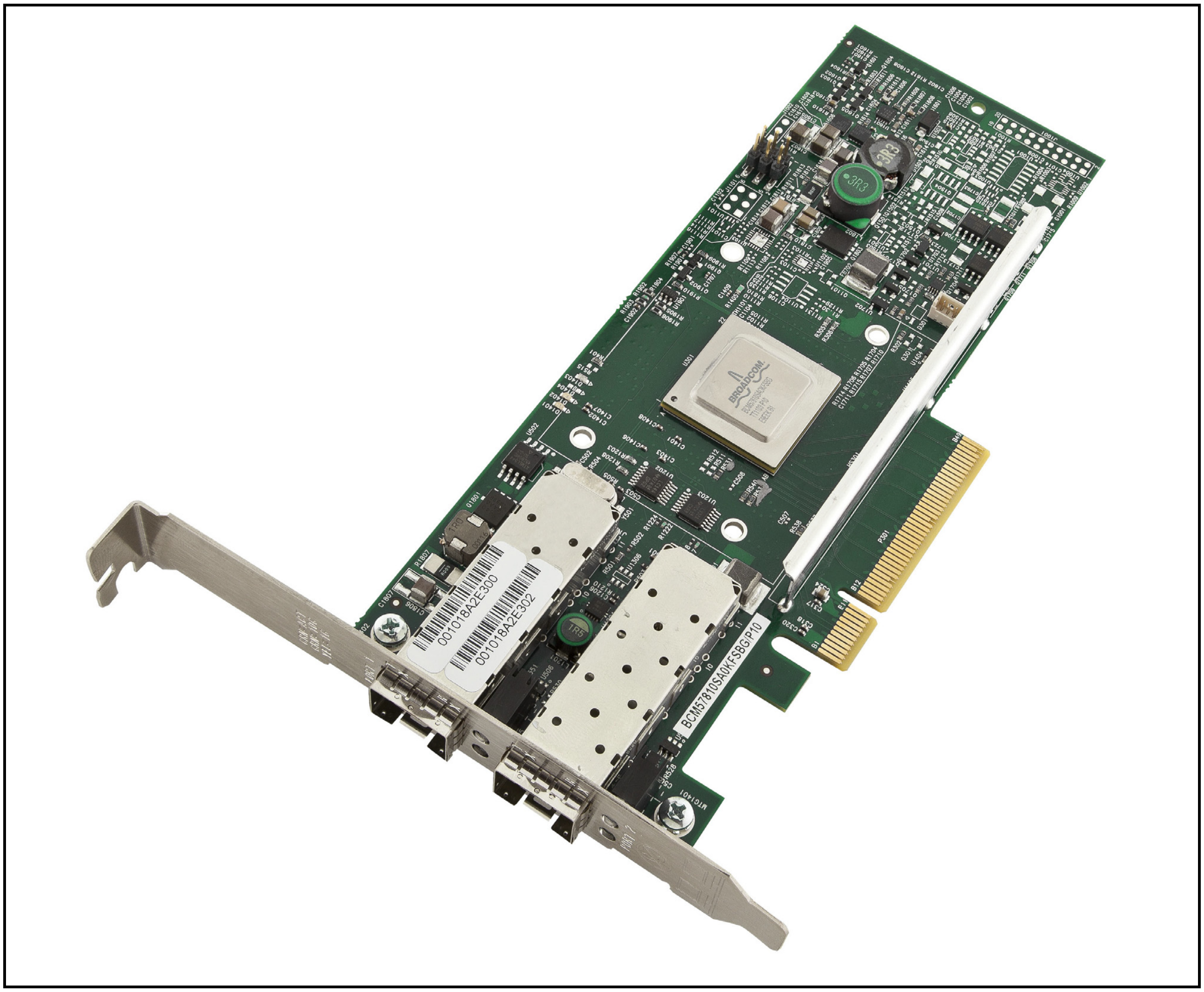


Figure 5: 57810S Converged NIC

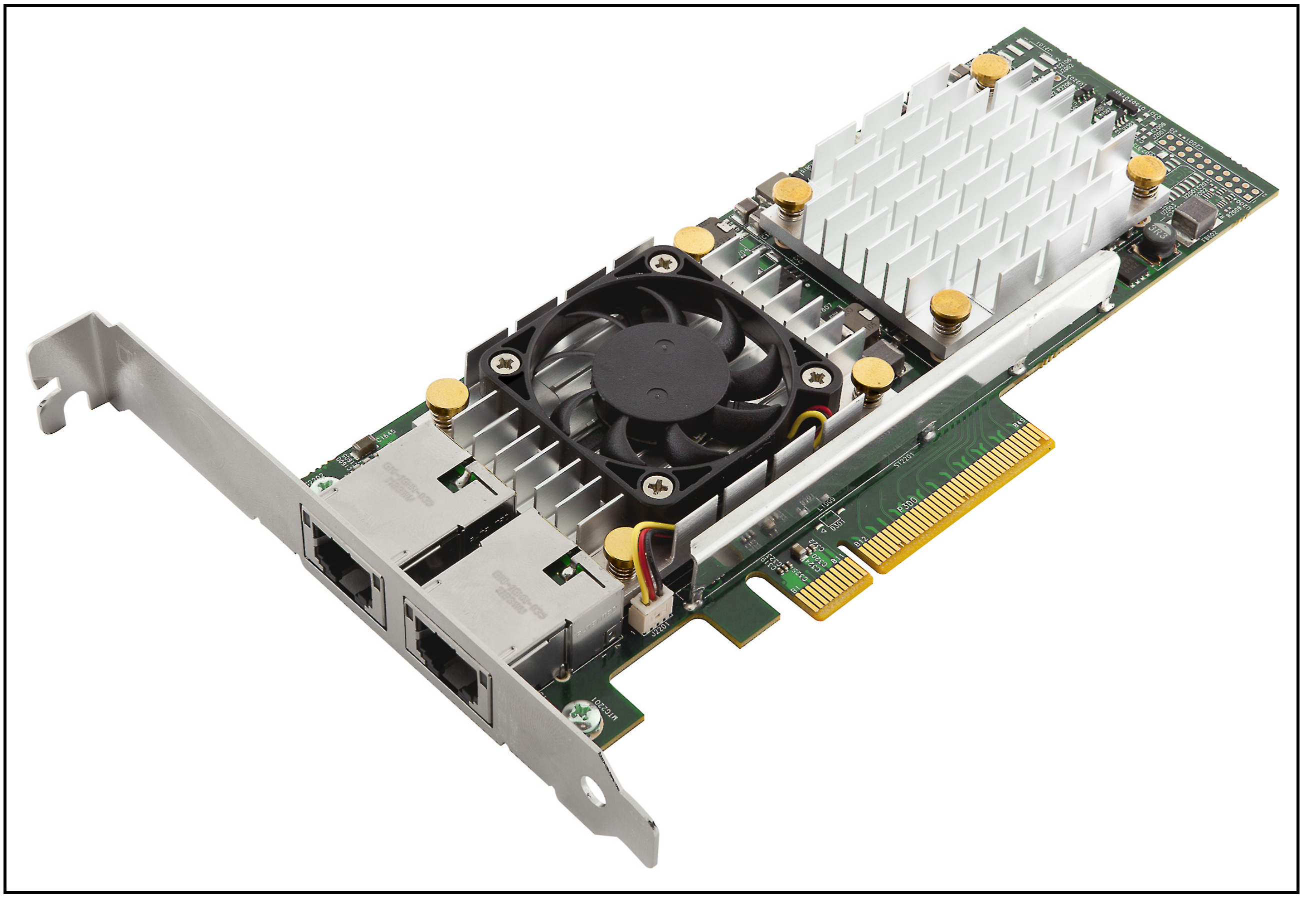


Figure 6: 57810S-T 10GBASE-T Converged NIC

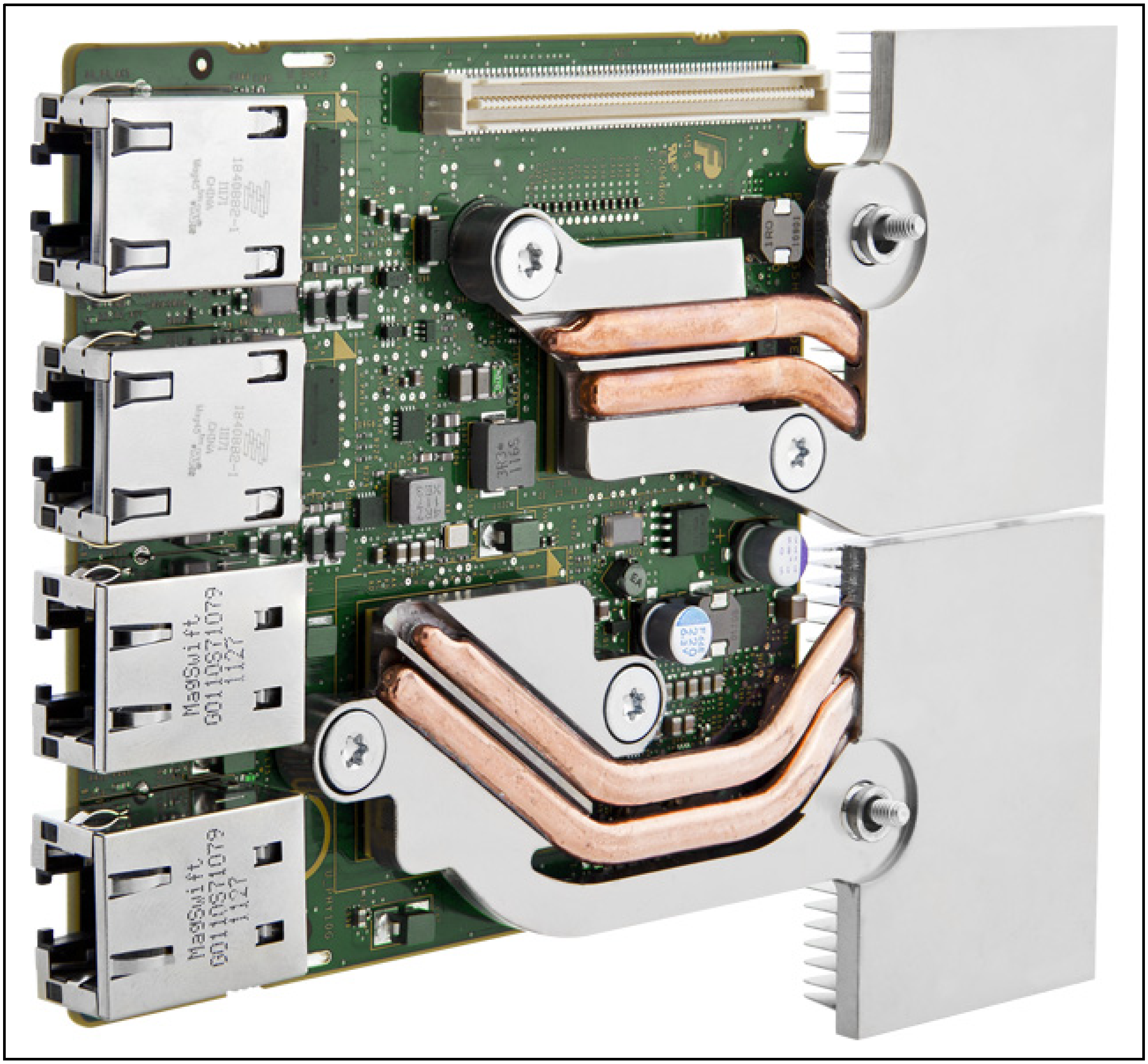


Figure 7: 57800S-T 10GBASE-T Rack Converged NDC

Audience

This document is written for the network administrator who wishes to partition the Broadcom network controller on a Microsoft® Windows Server, VMWare® ESX/ESXi, Oracle® Solaris, Citrix Xen Server, SUSE Linux Enterprise Server (SLES), and Red Hat Enterprise Linux® (RHEL) system with:

- up to eight functions (four per port) Ethernet enabled in addition to:
    - up to four functions (two per port) iSCSI HBA enabled (in operating systems where the specific HBA can be enabled).

  or

- up to two functions (one per port) FCoE HBA enabled plus up to two functions (one per port) iSCSI HBA enabled (in operating systems where the specific HBA can be enabled).

Acronyms and Abbreviations

In most cases, acronyms and abbreviations are defined on first use.

For a comprehensive list of acronyms and other terms used in Broadcom documents, go to: http://www.broadcom.com/press/glossary.php.

Technical Support

Broadcom provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates through its customer support portal (https://support.broadcom.com). For a CSP account, contact your Sales or Engineering support representative.

In addition, Broadcom provides other product support through its Downloads & Support site (http://www.broadcom.com/support/).

Configuring Switch Independent Partitioning

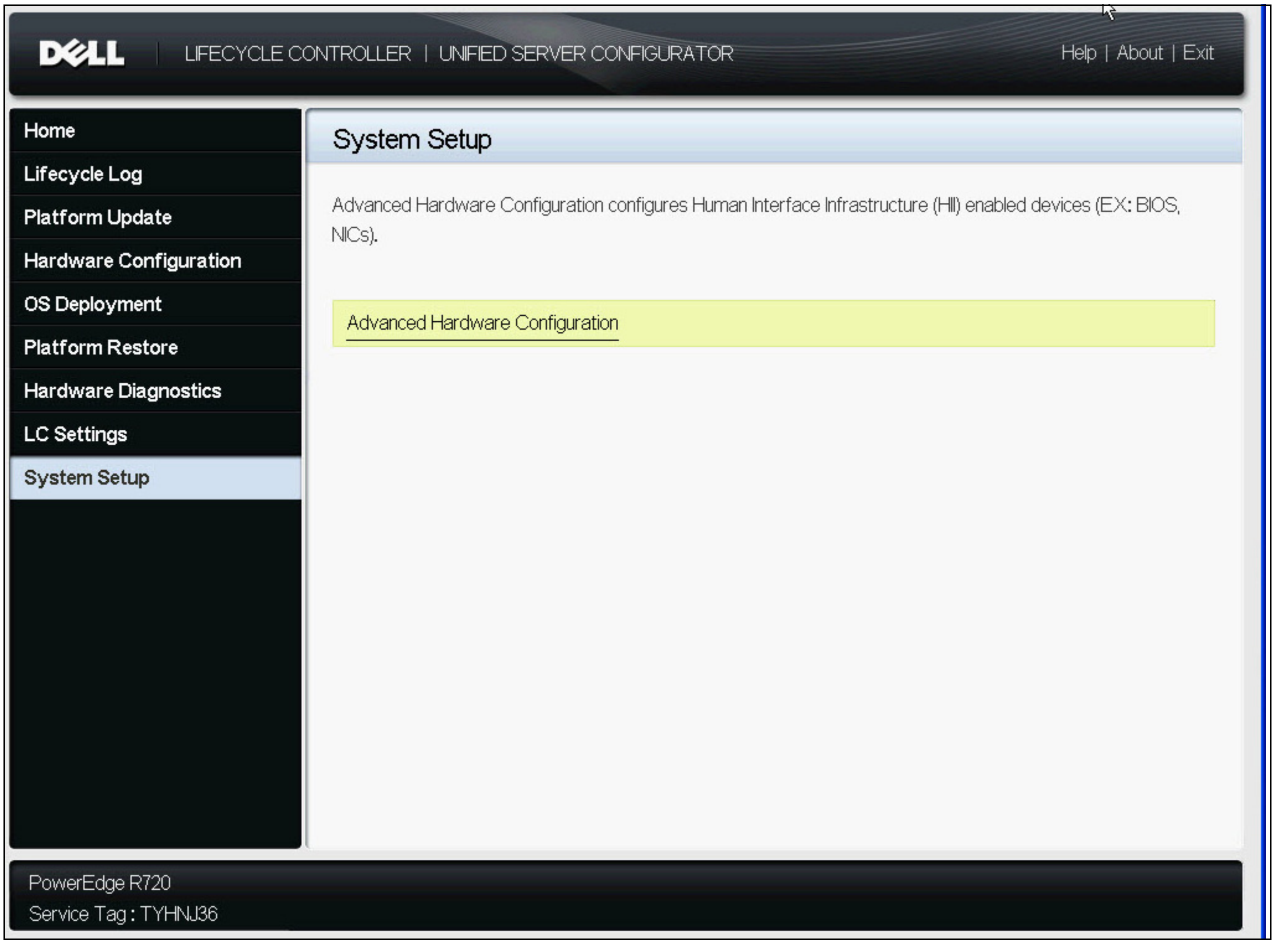
Using the Unified Server Configurator

Use Dell's Unified Server Configurator (USC) to configure Broadcom's device Switch Independent Partitioning parameters.

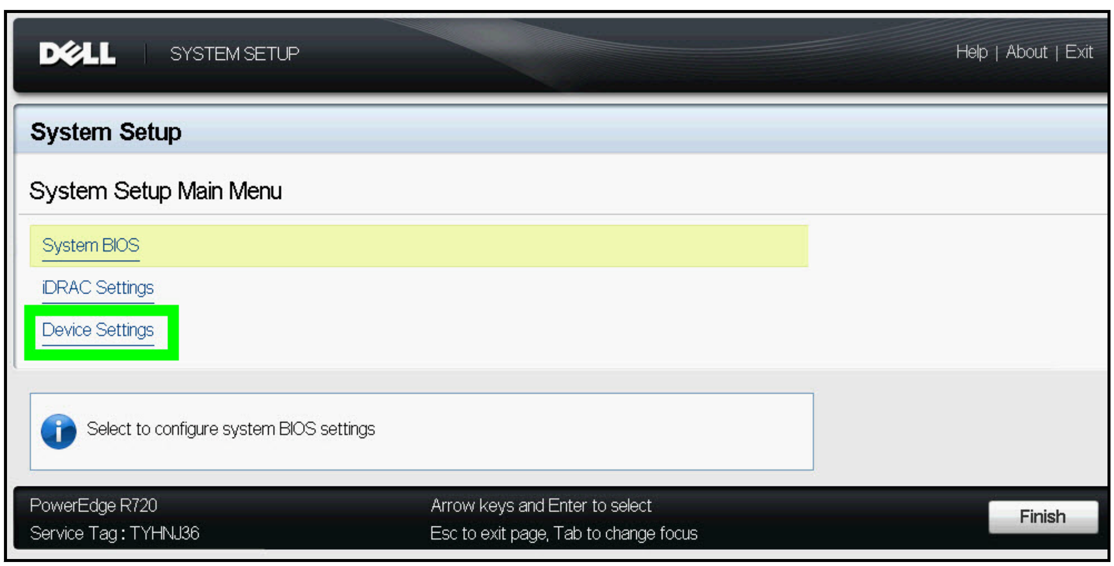
To configure Switch Independent Partitioning with the USC

1. Enter USC during system boot up by selecting the UEFI boot option. See the Dell website for more information on USC.

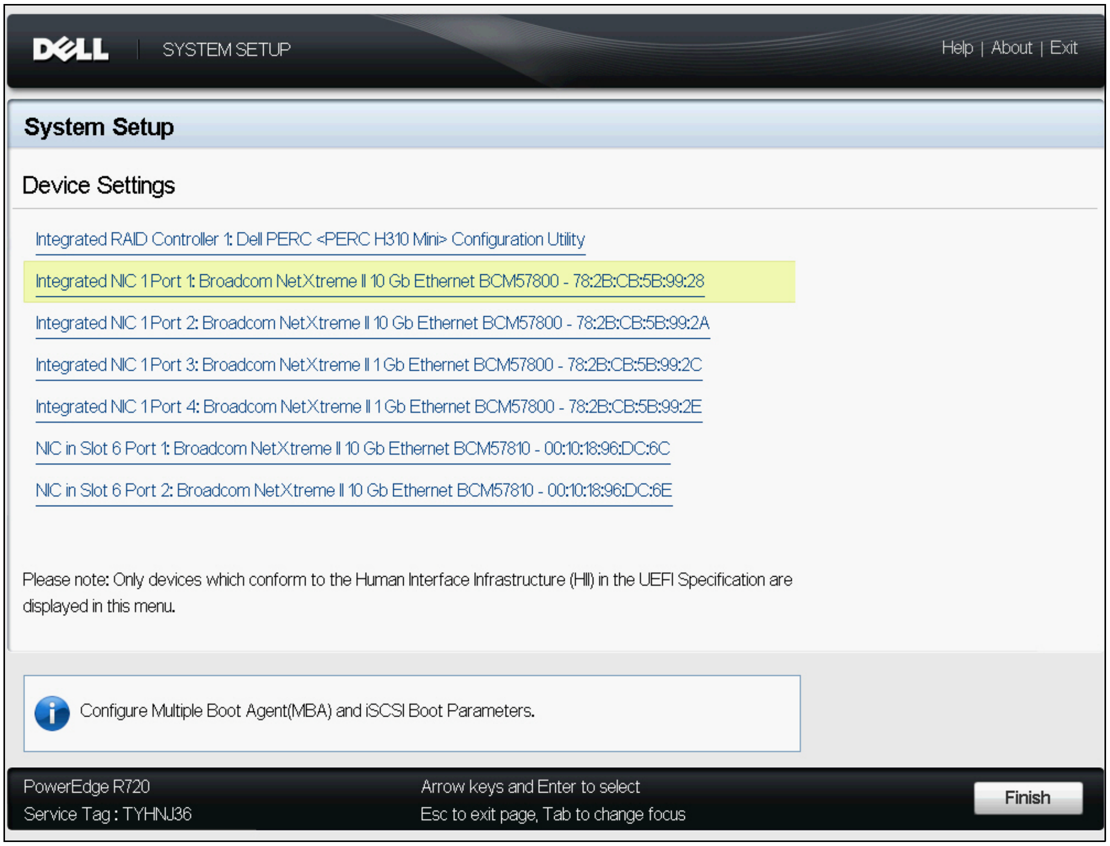
2. From the USC, select the **System Setup–Advanced Hardware Configuration** option.



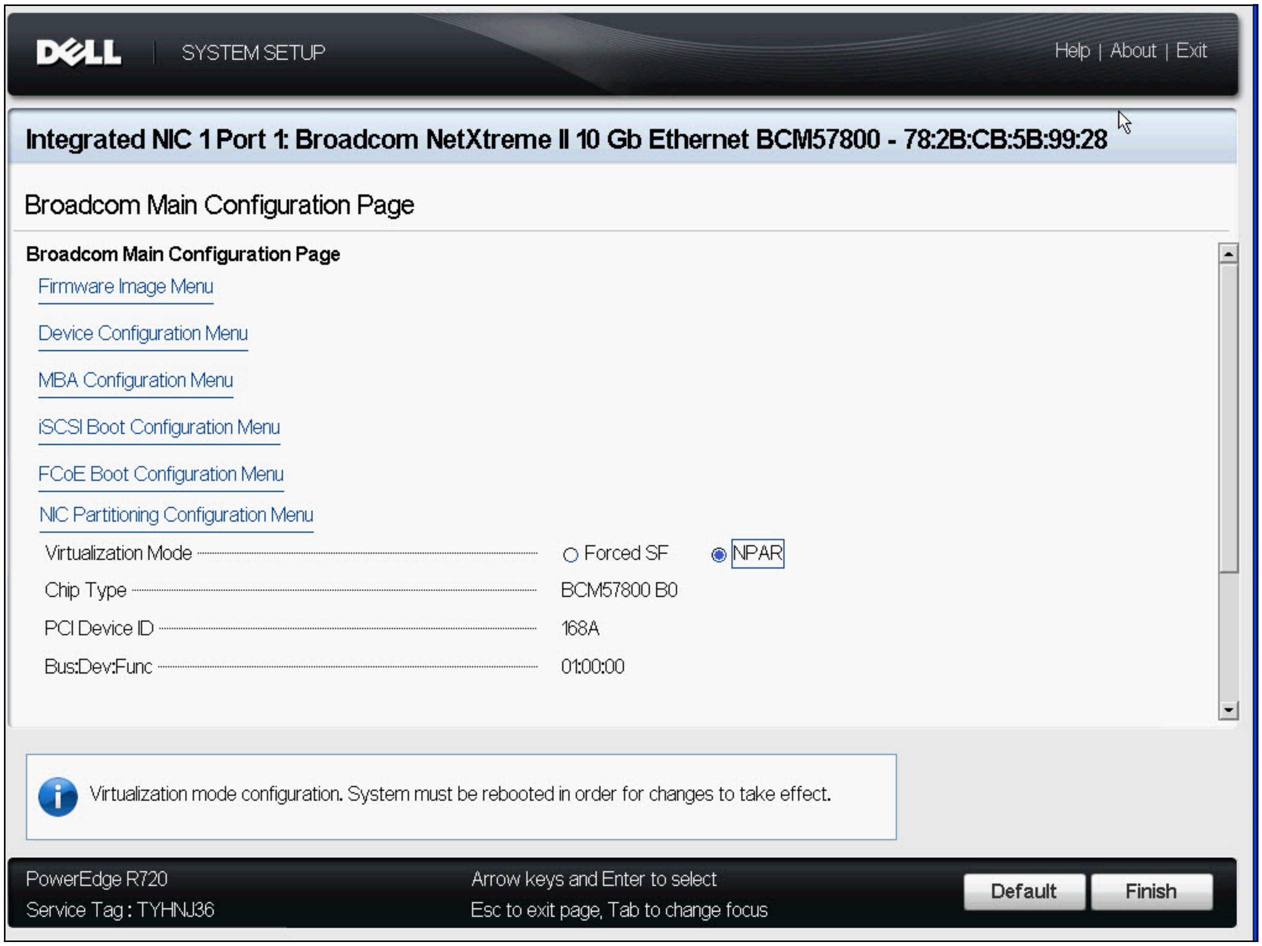
3. Select the System Setup's Device Settings option.



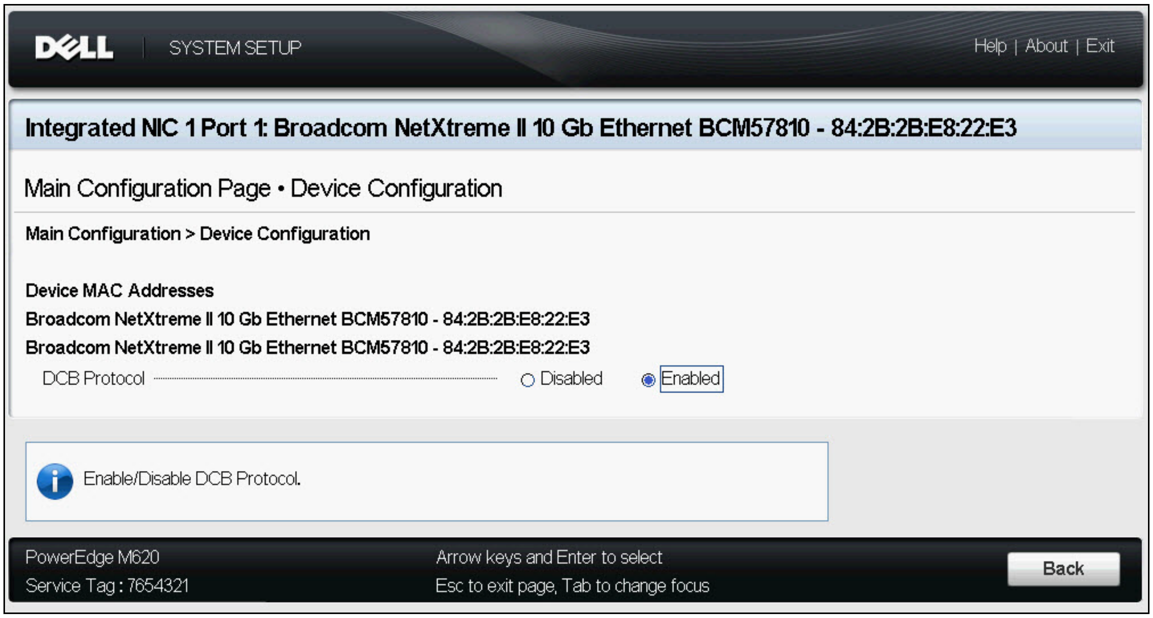
4. All of the Broadcom Ethernet Controller devices should be displayed on this page. Select the desired device port from the displayed list.



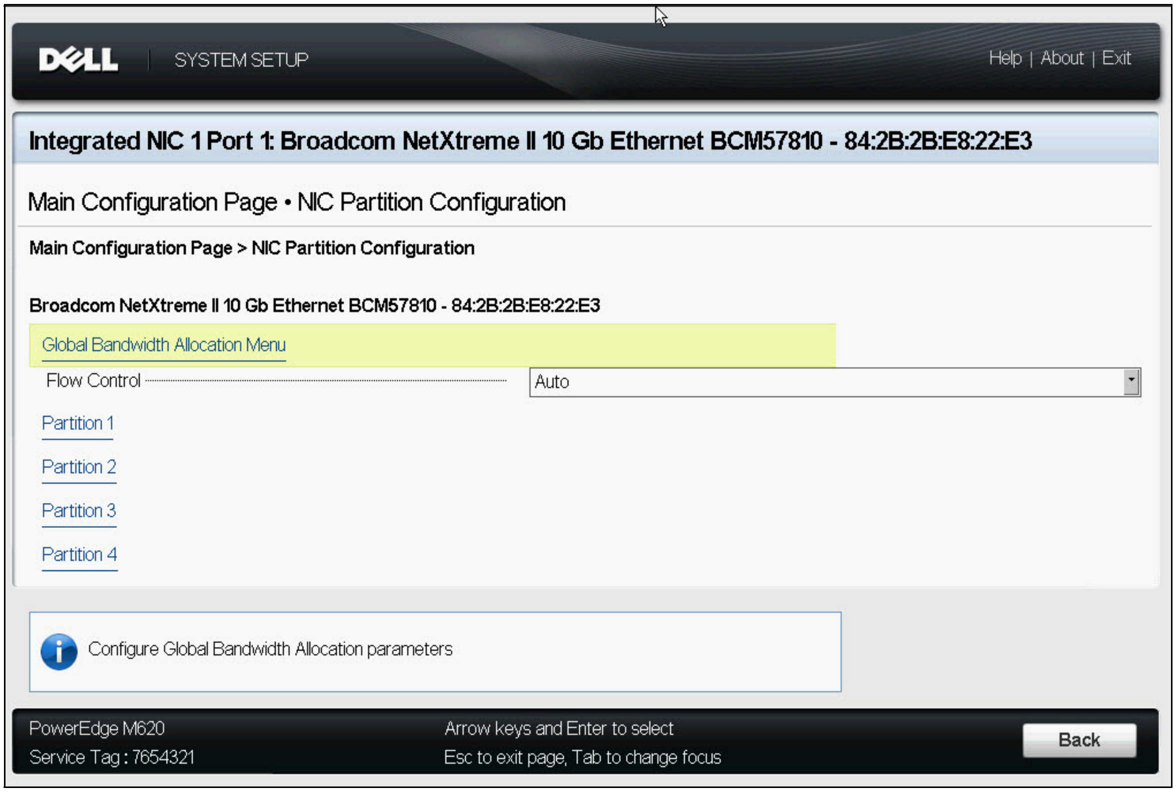
5. From the **Broadcom Main Configuration Page**, select the **Virtualization Mode Control** to turn on (NPAR) or off (Forced SF) the Switch Independent Partitioning mode of operation for the selected devices. Select the **Device Configuration Menu** option to enable or disable DCB mode. Select the **NIC Partitioning Configuration Menu** option to change the device's Switch Independent Partitioning settings—the mode must be enabled (in this window) before the option will appear (also in this same window).

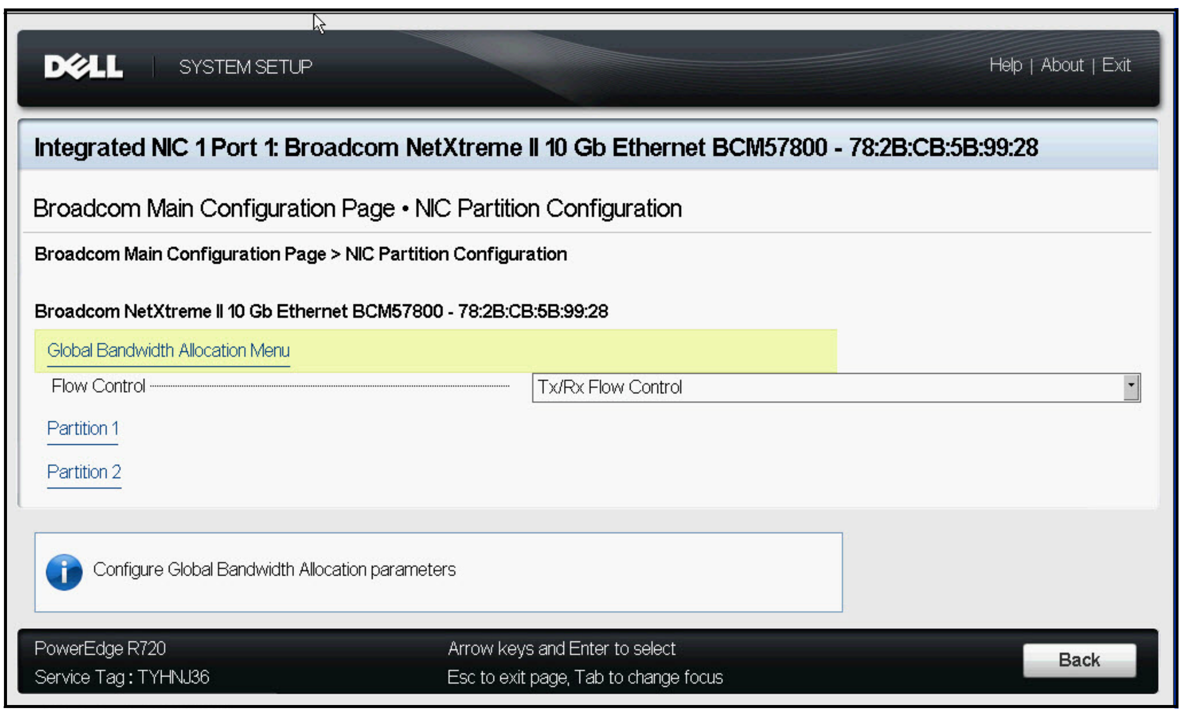


6. From the **Device Configuration** window, select either **Enabled** or **Disabled** for the device's DCB Protocol mode.

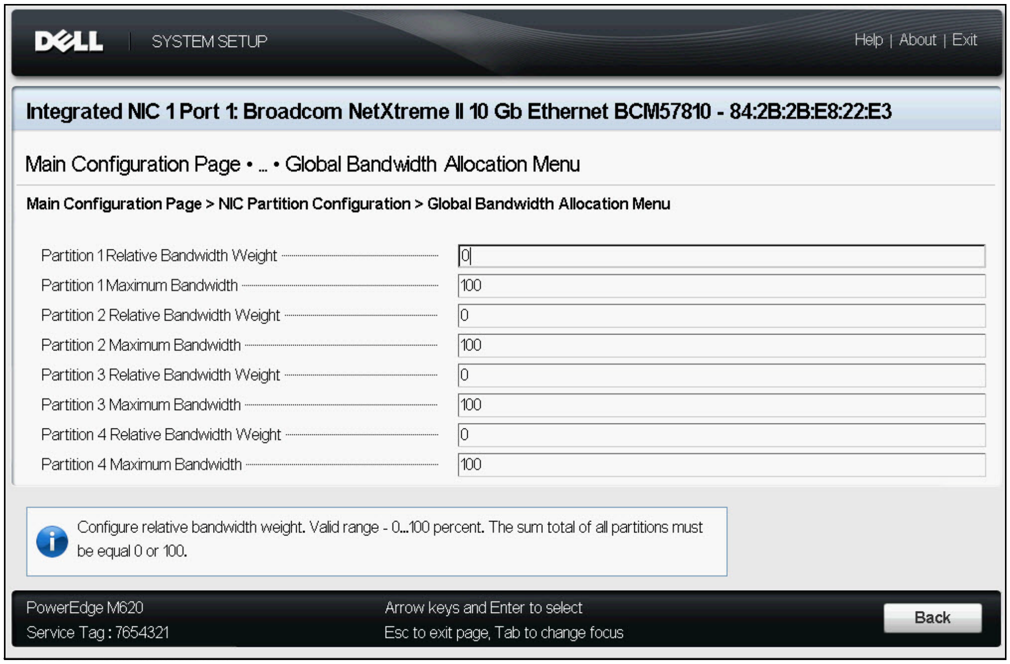


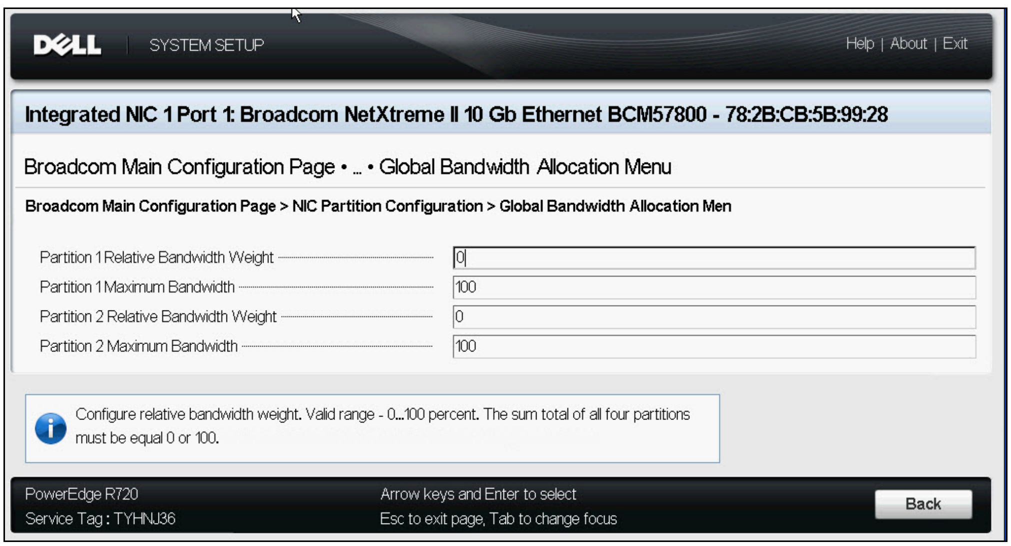
7. If the **NIC Partition Configuration Menu** is selected, instead of the **Device Configuration Menu** in Step 5, the device port's **Global Bandwidth Allocation Menu** option, **Flow Control** and individual **Partition** options are presented. The **Global Bandwidth Allocation Menu** option will open the control window for the Relative Bandwidth Weight (similar to a minimum bandwidth) and Maximum Bandwidth settings for that port's partitions. The **Flow Control** options sets this port's flow control settings. The four (on the 57712-k and 57810's) and two (on the 57800's) **Partition** options open a window that allows the enabling of available protocols for the selected partition and also displays the various MAC address information of that partition.





8. If the **Global Bandwidth Allocation Menu**, option was selected in Step 7, then the four (on the 57712-k/57810's) or two (on the 57800's) Relative Bandwidth Weights and Maximum Bandwidth setting window is presented.





See "Broadcom Advanced Control Suite 4 (BACS4)" on page 35 for more information on how BACS4 can also be used to control these settings.

The **Relative Bandwidth Weight** is the value the port gives to that single partition's send or outgoing traffic with respect to any other actively sending partitions on that port when there is more send traffic pending on the partitions than send bandwidth available on that port. It is more than just a minimum bandwidth setting. This setting follows these rules:

- The individual configurable value range is 0 to 100. The recommended Relative Bandwidth Weight value range is either all 0% OR no lower than 10% on a single partition.
- The **SUM** of a single port's partitions values **MUST** be either exactly **100** or exactly **0** (which means all four

of the partitions are set to 0).

- If one or more of a partition's weight is set to **0**, but the sum is **100** (i.e. not all of the partitions are set to zero) then that partition's relative bandwidth weight value is effectively **1** with respect to allocation calculations.
- Setting all the partition's values to **0** will give every traffic flow on every partition equal access to the ports available bandwidth without regard to which partition they are on unless restricted by the partition's Maximum Bandwidth settings.
- If the sum of the relative bandwidth weights is **100** and there is more than one type of traffic flow on a specific partition (i.e. iSCSI and L2 Ethernet or FCoE and L2 Ethernet) then the traffic on that specific partition will share the bandwidth being allocated as if there was only one traffic flow on that partition.
- The weight applies to all enabled protocols on that partition.
- The Relative Bandwidth Weight is not applicable when in Data Center Bridging (DCB) mode. In DCB mode, all traffic flows act as if their Relative Bandwidth Weights are set to all 0s. The DCB's Enhanced Transmission Selection (ETS) values are used instead.
- The Switch Independent Partitioning transmit direction traffic flow rates are affected by the three main modes in the following ways:
  - In non-DCB mode where the sum of the partition's Relative Bandwidth Weights equal 100, each partition's combined traffic flow is equally scheduled to transmit within the limitations of the partition's Relative Bandwidth Weight and Maximum Bandwidth settings and the overall connection's link speed. This means a specific partition's Relative Bandwidth Weight value will restrict the traffic flows sharing that partition's bandwidth allocation, as if one combined traffic flow with respect to the other actively sending partitions. The partition's send flow rate is based on the ratio of that partition's individual weight verses the aggregated weights of all the other actively sending partitions. Furthermore, each partition's combined traffic flow will be capped by that partition's Maximum Weight setting. See the User Guide's examples for more details. The actual inter-partition ratio of the two sharing traffic flows is controlled by the host OS. Think of the dynamic weight ratio as a variable sized funnel that could be further restricted by the Maximum Bandwidth fixed sized funnel with the OS determining how the sharing traffic types are pouring into the combined funnels.
  - In non-DCB mode where the sum of the partition's Relative Bandwidth Weights equals zeros (i.e., each partition's Relative Bandwidth Weight is set to zero), each individual traffic flow (i.e. Ethernet or iSCSI Offload or FCoE Offload) is equally scheduled to transmit within the limitations of the partition's Maximum Bandwidth and the overall connection's link speed. This means if the Maximum Bandwidth of a specific partition is set to less than 100%, then the traffic flows sharing that partition will be further restricted to where their combined traffic flow bandwidth will be capped by that per partition setting. If all four partition's individual Maximum Bandwidths are set to 100% (i.e. they are unrestricted), then each actively sending traffic flow (without regard to which partition they are on) will equally share the transmit directions total bandwidth (i.e. TX link speed). The actual inter-partition ratio of the two sharing traffic flows is controlled by the host OS. Think of the Maximum Bandwidth as a fixed sized funnel with the OS determining how the two sharing traffic types are pouring into that funnel.
  - In DCB mode, all of the partition's Relative Bandwidth Weights are disregarded and the individual traffic flows are scheduled to transmit within the limitations of the Priority Group's ETS value (determined by it's Traffic Type) and each partition's Maximum Bandwidth setting and the overall connections link speed. For example, the FCoE traffic type could be assigned to Priority Group 1 (PG1) and all of the other traffic types (iSCSI and Ethernet) could be assigned to another Priority Group (such as PG0). Each Priority Group has it's own ETS value (which works similarly to a minimum bandwidth

setting). DCB Lossless iSCSI (iSCSI-TLV) could be used simultaneously (in another Priority Group) or in place of FCoE for a similar effect where the Lossless iSCSI Offloaded traffic would go through it's assigned Priority Group while the Lossy Ethernet traffic would go through another. Similarly to the other two rate controlling modes, the host OS determines the actual inter-partition traffic ratio for the cases where two traffic types share the same partition.

Note: A traffic type's send flow rate will be approximately the ratio of its individual partition's relative bandwidth weight setting divided by the sum of the relative bandwidth weights of all the partitions currently actively sending on that port or that partition's maximum bandwidth setting, whichever is lower. In the case where the Relative Bandwidth Weights are all zeros OR in DCB mode, each traffic type will have an equal "weight" with respect to one another (see "Examples" on page 94).

Note: DCB mode is supported in Windows, Linux (RHEL v6.x and SLES11 SP1), Oracle Solaris 10 u9/u10, and VMWare ESXi 5.0 OS's on the 57712-k/578xxS devices. VMWare ESX/ESXi 4.1 does not support DCB (which includes both FCoE and DCB Lossless iSCSI) on the 57712-k/578xxS devices.

Each partition's **Maximum Bandwidth** settings can be changed in the same way and has a range of 1 to 100% in increments of 1% of the port's current Link Speed (at 10 Gbps this would be in ~100 Mbps increments and at 1 Gbps this would be in ~10 Mbps increments). This setting limits the most send bandwidth this partition will use and will appear as its approximate link speed in various places in the respective operating system even though the four partition's are sharing the same overall connection - i.e. the four partitions may advertise in the OS that their link speed is 10Gbps each, but they all share the same single 10Gbps connection. Displayed values may be rounded off by various applications. The Maximum Bandwidth value is applicable to both DCB and non-DCB modes of operation. The Maximum Bandwidth value is applicable to the send (TX) direction only.

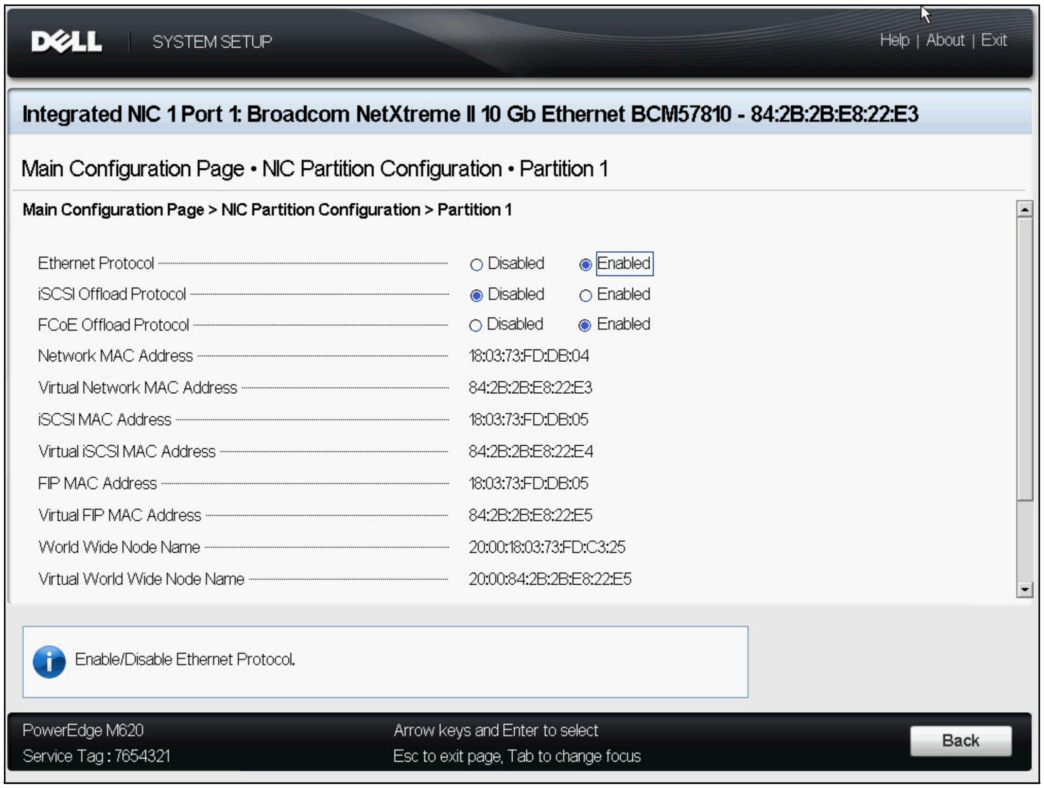
Note: A partition's send Maximum Bandwidth setting does not affect a partition's receive direction traffic bandwidth, so the link speed displayed for the partition is for the send/transmit/outgoing direction only. All partitions receive direction maximum bandwidth is always the ports current Link Speed and is regulated by the attached switch port just as it is in SF mode when multiple (L2 Ethernet and iSCSI Hardware Offload and FCoE Hardware Offload) traffic protocol types are enabled.

The Maximum Bandwidth settings can be used to "oversubscribe" a port. This is done by setting the four partitions of that single port to having a total Maximum Bandwidth setting **SUM** of more than 100% (i.e., 10000 Mbps or 10 Gbps). That just means the various partitions will attempt to take as much bandwidth as allowed (by their specific setting maximum limits and weights) as their individual traffic flow needs change. In an oversubscription situation, the 57712-k/578xxS device will ration out free bandwidth based on the weights (sum is 0 verses sum is 100) and maximum settings and the mode (DCB verses non-DCB) it is in. The above example shows the first port's four partitions being set to 10+10+80+80 = 180, which means the port is 180% subscribed (18 Gbps) or 80% oversubscribed (i.e., 18 Gbps subscribed – 10 Gbps line rate available = 8 Gbps oversubscribed). The Maximum Bandwidth setting applies to all protocols enabled on that partition.

Note: When Switch Independent Partitioning mode is first enabled or after a reset, the default values for all four partitions is **Relative Bandwidth Weight** = 0 and **Maximum Bandwidth = 25**.

See “Examples” on page 94 for more details on both the **Relative Bandwidth Weight** and **Maximum Bandwidth** settings affect traffic flow in DCB and non-DCB modes of operation.

9. If one of the **Partition** options was selected in the **NIC Partition Configuration** window in Step 5, you are able to change the selected partition's protocol settings here.



10. This window, selects which protocols it will support and also displays the partitions assigned Networking MAC address, the iSCSI Offload MAC address, the FCoE Offload FIP MAC address, the FCoE Offload Node WWN, and FCoE Offload Port WWN values. BACS4 can also be used to control and view these settings.

Protocol selection follows these rules:

- A maximum of two iSCSI or one FCoE and one iSCSI Offload Protocols (HBA) can be enabled over any two of the available partitions of a single port.
- The FCoE Offload Protocol is only available if DCB is also enabled and active on that port (i.e., the 57712-k/578xxS device port is connected to a DCB compliant and enabled link partner).
- The iSCSI Offload Protocol can function without DCB but if DCB Lossless iSCSI Offload TLV is required, then DCB must be enabled and active on that port (i.e., the 57712-k/578xxS device port is connected to a DCB compliant and enabled link partner).
- Only one Offload Protocol (either iSCSI or FCoE) can be enabled per single partition in Switch Independent Partitioning mode.
- For simplicity, using the first partition of a port for FCoE offload protocol is recommended since the FCoE port WWN will be the same for both SF and Switch Independent Partitioning mode on the same port. This will make your Fiber Channel Forwarder (FCF) switch configuration much simpler.
- For Windows operating systems, you can have the Ethernet Protocol enabled on all, some, or none of the

four partitions on an individual port simultaneously with any enabled offload protocols.

- For Linux OSs, the Ethernet protocol will always be enabled (even if disabled in USC).
- For simplicity, we recommend always using the first two partitions of a port for any iSCSI offload protocols.
- For Windows OSs, the Ethernet protocol does not have to be enabled for the iSCSI or FCoE offload protocol to be enabled and used on a specific partition.
- For VMWare ESX/ESXi 4.1, in Switch Independent Partitioning mode, the host and hosted Virtual Machines (VMs) can only connect to enabled Ethernet and iSCSI Offload protocol partitions.
- For VMWare ESXi 5.0, in both SF and Switch Independent Partitioning modes, the host and hosted VMs can connect to the enabled Ethernet and iSCSI Offload and FCoE Offload protocol enabled partitions.

Supported Operating Systems

The 57712-k/578xxS device SF and Switch Independent Partitioning mode supported operating systems are shown in Table 1.

Note: The drivers may not be in the box.

Table 1: Protocols Available in Operating Systems Versus SF and Switch Independent Partitioning Modes

	SF Mode			Switch Independent Partitioning Mode		
Operating System	***Ethernet***	***iSCSI Offload***	***FCoE Offload***	***Ethernet***	***iSCSI Offload***	***FCoE Offload***
Windows 2008[a]/R2[a]	Yes	Yes	Yes	Yes	Yes	Yes
Windows 2008 R2 Hyper-V[a]	Yes	Yes	Yes	Yes	Yes	Yes
Oracle Solaris 10 u9[a]/ u10[a]	Yes	No	Yes	Yes	No	Yes
RHEL5.5+	Yes	Yes	No	Yes	Yes	No
RHEL 6.0+[a]	Yes	Yes	Yes	Yes	Yes	Yes
SLES10	Yes	Yes	No	Yes	Yes	No
SLES11 SP1[a]	Yes	Yes	Yes	Yes	Yes	Yes
Citrix XenServer v6.0	Yes	No	No	Yes	No	No
VMWare ESX/ESXi 4.1	Yes	Yes	No	Yes	Yes	No
VMWare ESXi 5.0[a]	Yes	Yes	Yes	Yes	Yes	Yes

a. DCB (DCBX/PFC/ETS) supported.

Supported Server Platforms

The 57712-k and the new 12G release networking devices (5719/5720/578xxS) are available for a wide variety of Dell server platforms as shown in Table 2.

Table 2: Dell 12G Networking Device Release verses Dell Servers

	5719 NIC	*5720 NIC*	*5720 rNDC*	*5719 bMezz*	*57712-k bNDC*	*57800S rNDC*	*57810S-K bNDC*	*57810S-K bMezz*	*57810S NIC*	*57800S-T rNDC*	*57810S-T NIC*
Dell Servers	R720, R720XD, R620, T620 and other rack and tower servers	R720, R720XD, R620, T620 and other rack and tower servers	R720, R720XD and R620 rack servers	M620 blade server	M710HD and M915 blade servers	R720, R720XD and R620 rack servers	M620 blade server	M620, M910 and M915 blade servers	R720, R720XD, R620, T620 and other rack and tower servers	R720, R720XD and R620 rack servers	R720, R720XD, R620, T620 and other rack and tower servers
Blade IO Modules	NA	NA	NA	1GbE Switches and PTMs	PTM-k, M8024-k and M8428-k	NA	PTM-k, M8024-k and M8428-k	PTM-k, M8024-k and M8428-k	NA	NA	NA
Ports	4	2	4	4	2	4	2	2	2	4	2
Interface	1000BASE-T	1000BASE-T	1000BASE-T	1000BASE-T	KR	2x10G (SFP+) 2x1000BASE-T	KR	KR	10G (SFP+)	2x10GBASE-T 2x1000BASE-T	10GBASE-T

Viewing and Configuring the Partitions

- Windows Server 2008 R2
- Red Hat Enterprise Linux v6.x
- VMWare ESX/ESXi 4.1
- VMWare ESXi 5.0

Windows Server 2008 R2

Installing the Latest Dell Drivers

When the 57712-k/578xxS device is first installed, the iSCSI and FCoE devices may not appear. If the latest Dell driver is already present on the system, the 57712-k/578xxS device will be identified and the NDIS personality/ protocol will be installed. If the latest NetXtreme II drivers are not present on the system, go to the Dell driver download web site (http://support.dell.com/support/downloads/ under the specific Dell blade server platform) and download/install the latest NetXtreme II network drivers (software version 17.2.x or later and firmware version 7.2.x or later) for your specific server system.

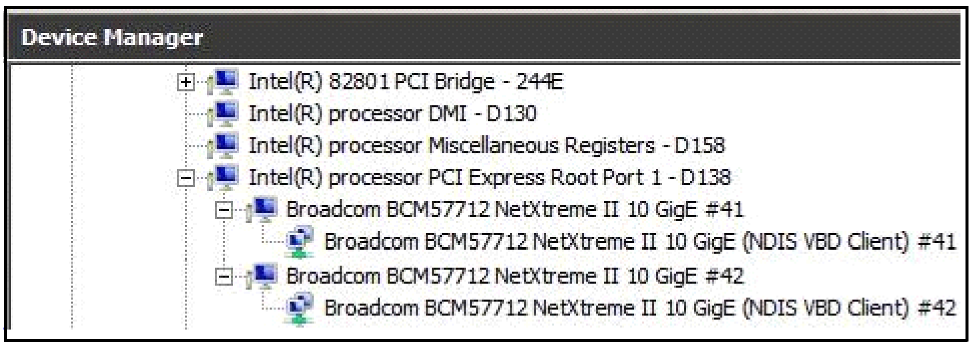


Figure 8: Windows Device Manager

To enable the applicable offload devices to be detected by the operating system, start BACS4 and while in 57712-k/578xxS SF mode, select the port **System Device>Configuration>Resource Reservations**, check the boxes to enable the applicable iSCSI and FCoE Offload Engines protocols and then click **Apply**. Click **Yes** when the temporary network connection interruption warning displays and wait for the discovered devices to be installed by Windows — no reboot is necessary while in SF mode. This must be done for each applicable device port.

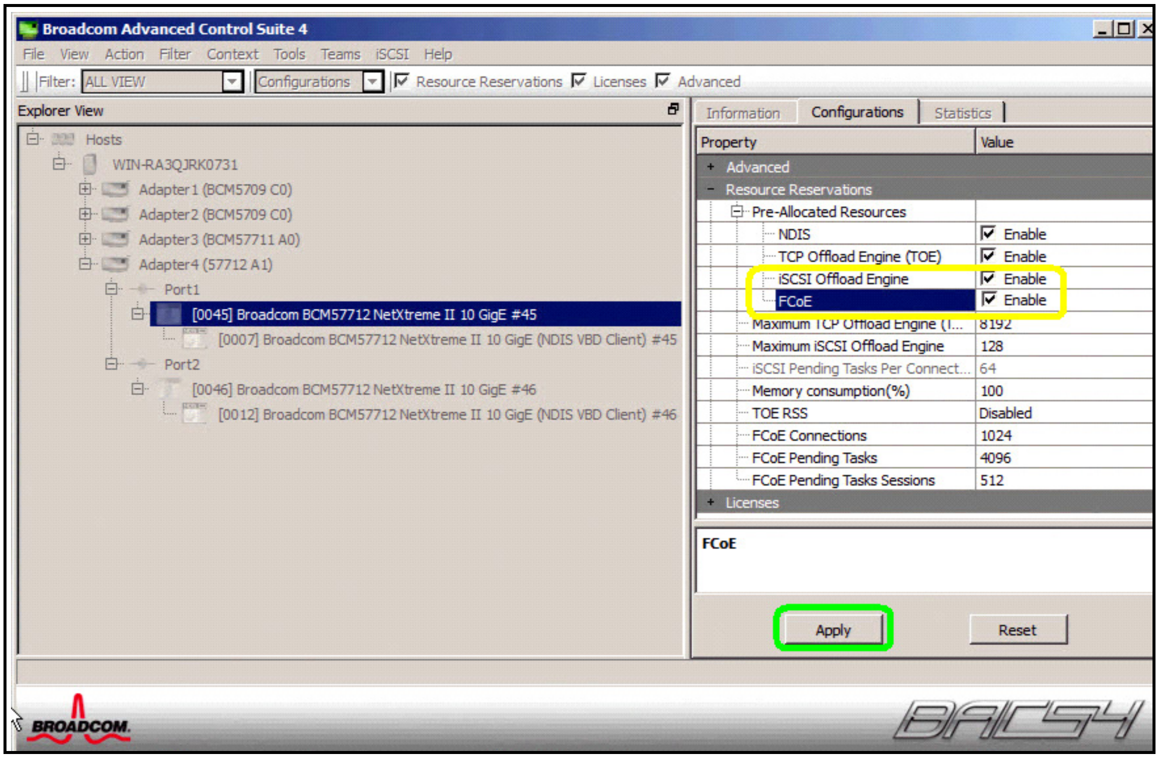


Figure 9: Broadcom Advanced Control Suite 4

If you are in Switch Independent Partitioning mode, go to the **57712-k/578xxS Adapter>Configuration>Property** window and click the **+** next to the NIC Partition to expand the menu. In the expanded menu, if the NIC Partition setting is unchecked (**Disabled**), change it to checked (**Enabled**) and then set the desired partition's Ethernet/NDIS, iSCSI and FCoE protocols. Also set the Relative Bandwidth Weights, Maximum Bandwidth settings and then click **Apply**. You must reboot the system for Windows to discover and install the device drivers.

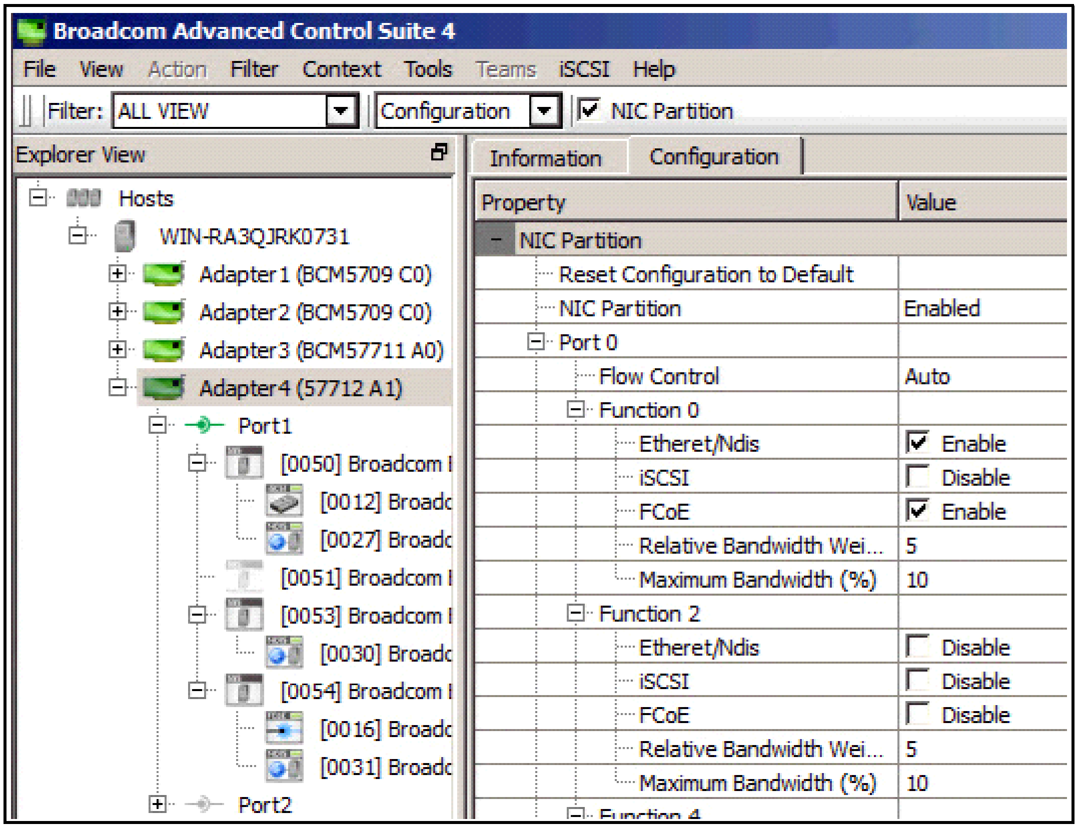


Figure 10: Broadcom Advanced Control Suite Adapter Settings

After the devices are installed, the enabled devices (L2 Ethernet NDIS, FCoE and iSCSI) will be visible in the Windows Device Manager and BACS4. The following is the 57712-k's Device Manager's display in SF mode (see Figure 11). The 578xx devices are displayed similarly.

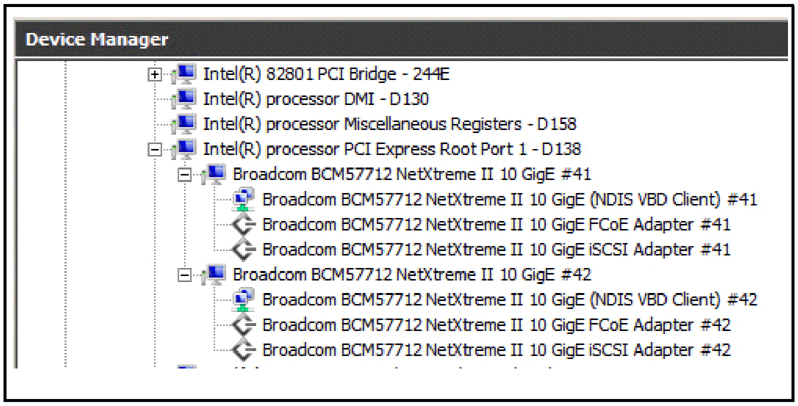
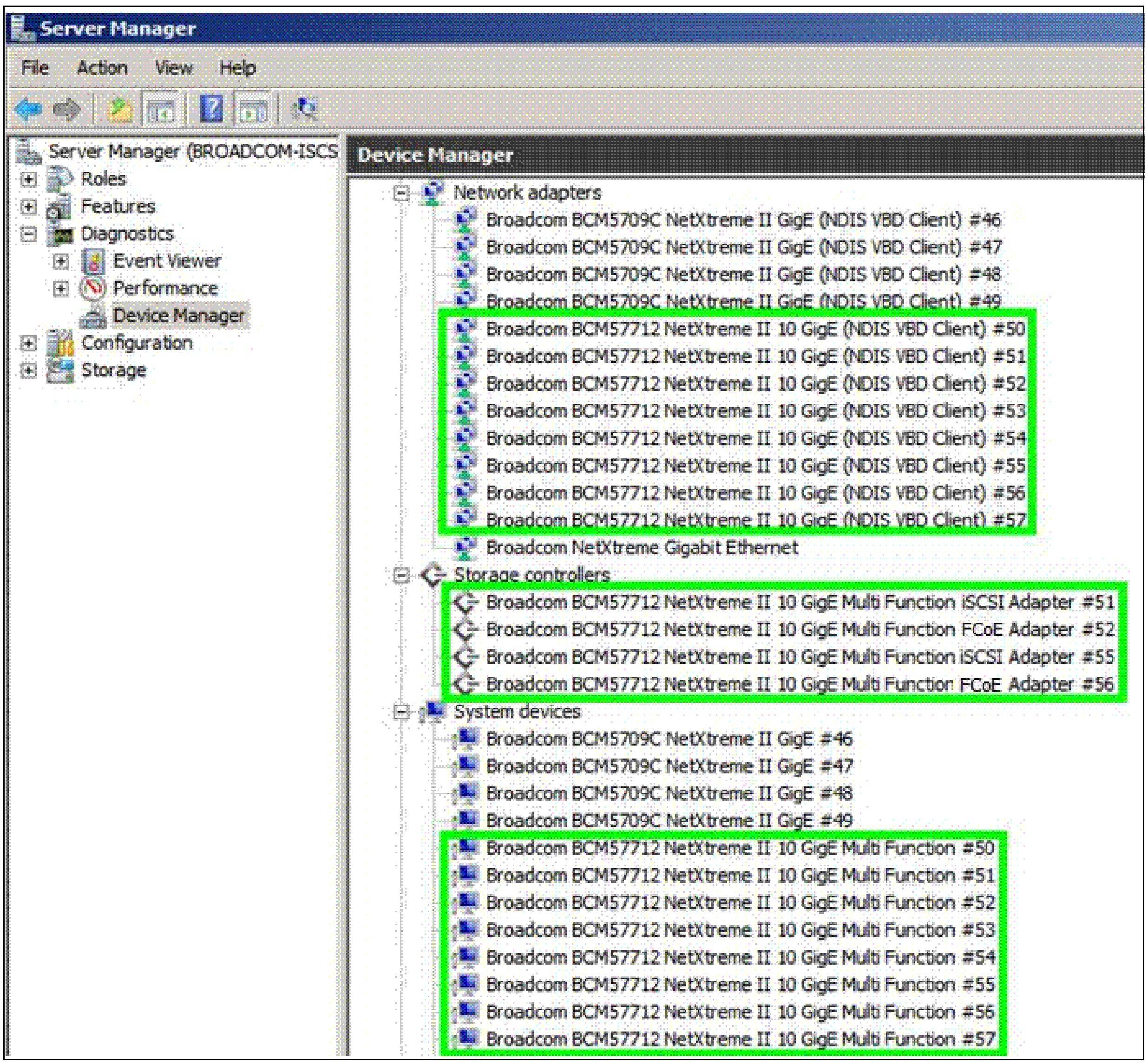


Figure 11: Windows Device Manager

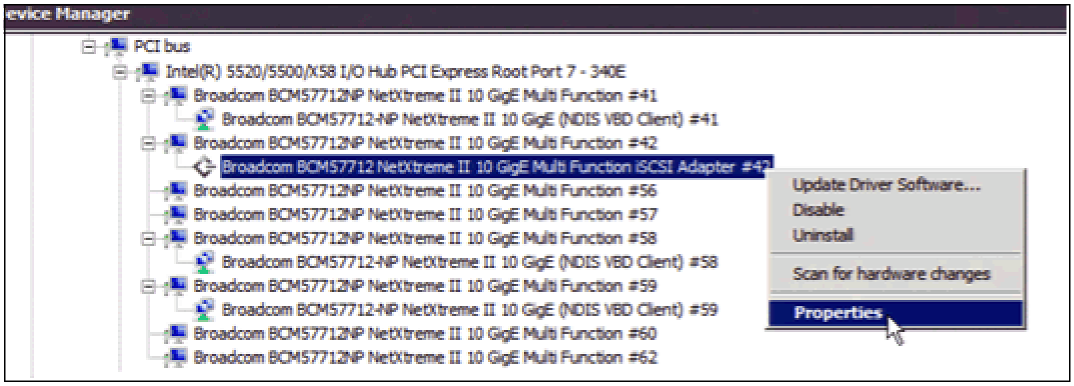
Viewing the Enabled Devices in Device Manager

Windows shows all of the enabled devices in Device Manager with the respective USC/BACS4-enabled Switch Independent Partitioning protocols. The following example shows a 57712-k but the 578xx devices would be similar:

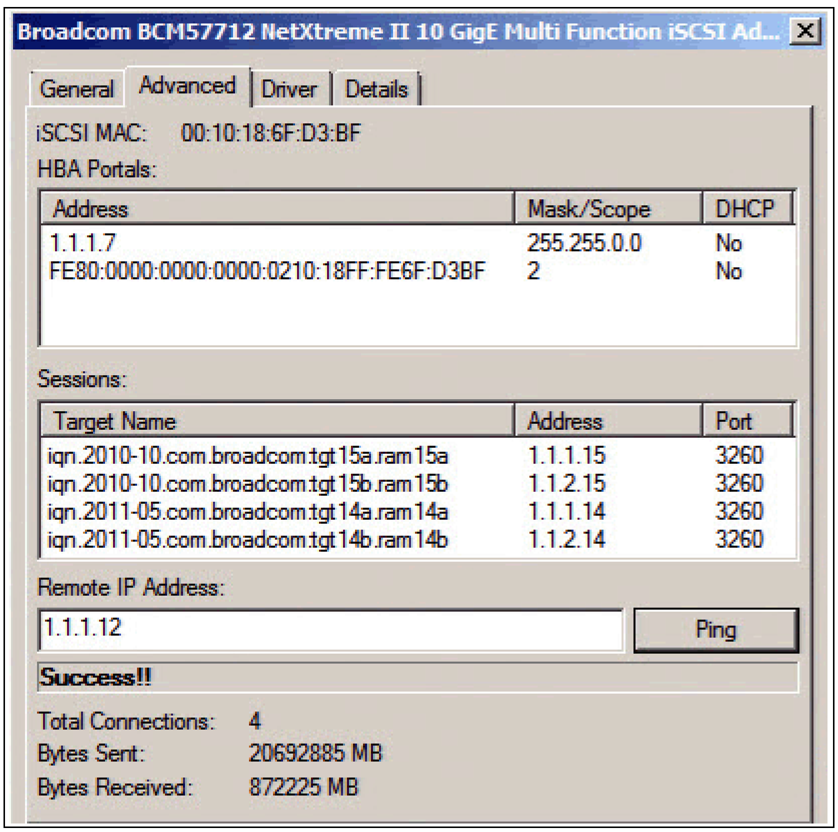
- Eight-enabled Ethernet protocol partitions (four possible per port) as the Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #50 through #57 in the Network Adapters section.
- Two enabled iSCSI protocol partitions (up to two are possible per port if no FCoE is enabled) as the Broadcom BCM57712 NetXtreme II 10 GigE Multifunction iSCSI Adapters #51 and #55 AND two enabled FCoE protocol partitions (one possible per port) as the Broadcom BCM57712 NetXtreme II 10 GigE Multifunction FCoE Adapters #52 and #56 in the Storage Controllers section
- Eight Broadcom BCM57712 NetXtreme II 10 GigE Multifunction virtual bus devices #50 through #57 in the System Devices section. These eight virtual bus system devices are always present and are not controlled by what protocol is enabled in USC.



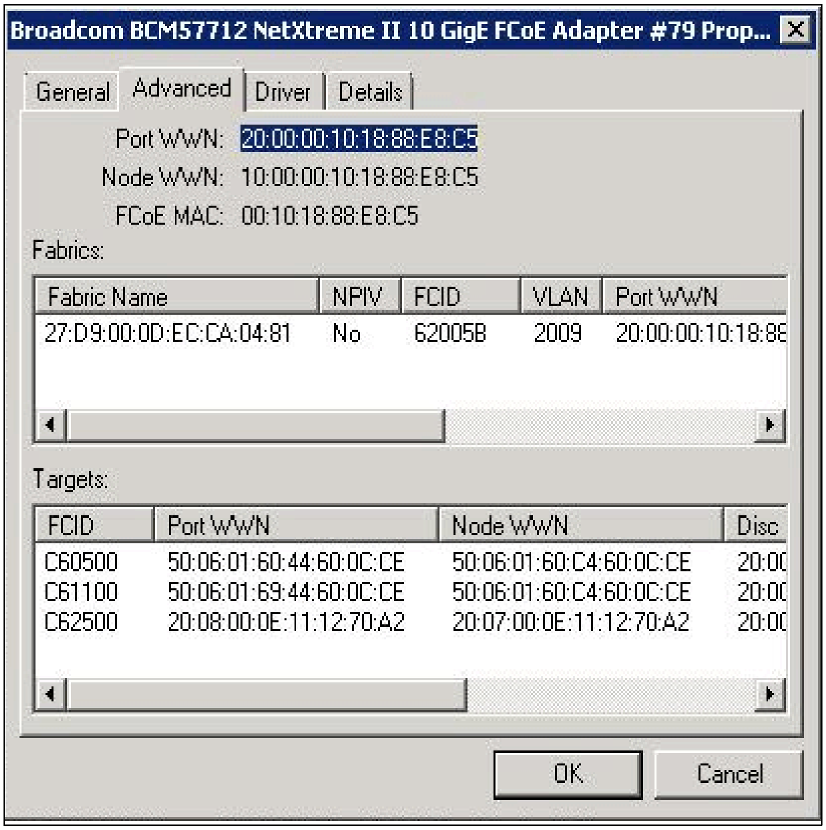
Right click the specific device and select its **Properties** to access some of the advanced features of the device.



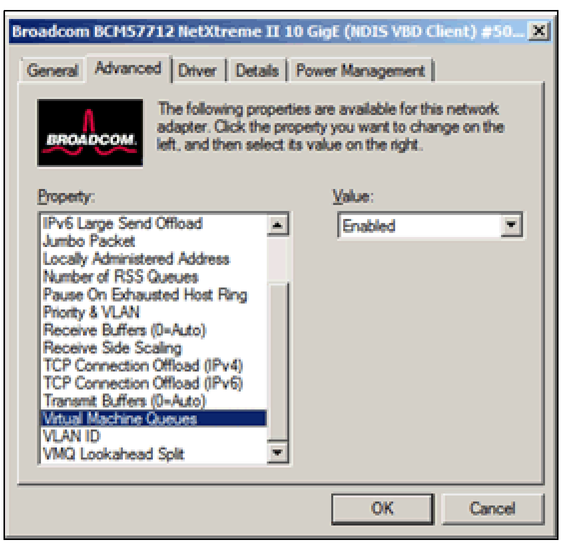
This brings up that device's properties window. The following is the BCM57712 iSCSI device showing the HBA and connected target session information, send/receive statistics and ping test results.



The following shows the properties window for the FCoE device showing World Wide IDs, connected Fabric and Target information.



The following shows the property window for the NDIS (Ethernet) device:



Note: In Switch Independent Partitioning mode, MS Window's TCP Chimney Offload or TOE functionality can be enabled or disabled on a per partition granularity in this Advanced Properties control window and in BACS4's NDIS Advanced Properties control window.

The number of iSCSI connections that are being offloaded by the hardware can be seen by using the “iscsicli sessionlist” DOS command, which lists the Initiator Name with an iSCSI offloaded connection entry beginning with “B06BDRV...” and a non-offloaded connection beginning with “Root...”.

The number of currently active Windows TOE connections can be viewed by using the “`netstat -not`” command in a DOS window.

```
C:\Users\Administrator>netstat -not

Active Connections

  Proto  Local Address          Foreign Address        State           PID
Offload State

  TCP    1.1.1.24:49301         1.1.1.25:5010          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49302         1.1.1.25:5012          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49303         1.1.1.25:5001          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49304         1.1.1.25:5013          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49305         1.1.1.25:5008          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49306         1.1.1.25:5003          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49307         1.1.1.25:5014          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49308         1.1.1.25:5000          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49309         1.1.1.25:5004          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49310         1.1.1.25:5006          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49311         1.1.1.25:5009          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49312         1.1.1.25:5015          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49313         1.1.1.25:5002          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49314         1.1.1.25:5011          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49315         1.1.1.25:5005          ESTABLISHED     3352
Offloaded
  TCP    1.1.1.24:49316         1.1.1.25:5007          ESTABLISHED     3352
Offloaded
  TCP    3.3.3.24:49289         3.3.3.24:49293         ESTABLISHED     3716
InHost
  TCP    3.3.3.24:49291         3.3.3.24:49294         ESTABLISHED     3364
InHost
  TCP    3.3.3.24:49293         3.3.3.24:49289         ESTABLISHED     2120
InHost
  TCP    3.3.3.24:49294         3.3.3.24:49291         ESTABLISHED     2120
InHost
```

Broadcom Advanced Control Suite 4 (BACS4)

The BACS4 utility provides useful information about each network adapter that is installed in your system, including partitioned adapters. BACS4 enables you to perform detailed tests, diagnostics, and analyses, as well as allows you to view and modify various property values and view traffic statistics for each adapter, including other vendor devices.

BACS4 allows the enabling and configuring of both ports Switch Independent Partitioning flow control/ protocols/Relative Bandwidth Weights/Maximum Bandwidth settings.

The following figure shows the per partition Switch Independent Partitioning settings (see Figure 12). This is where BACS4 can enable or disable Switch Independent Partitioning mode. This is also where BACS4 controls the Switch Independent Partitioning per port IEEE 802.3x Link-Level **Flow Control** settings (used when DCB's PFC is disabled), enabled protocols (Ethernet or iSCSI or FCoE), the **Relative Bandwidth Weight** values, and the **Maximum Bandwidth** values per partition.

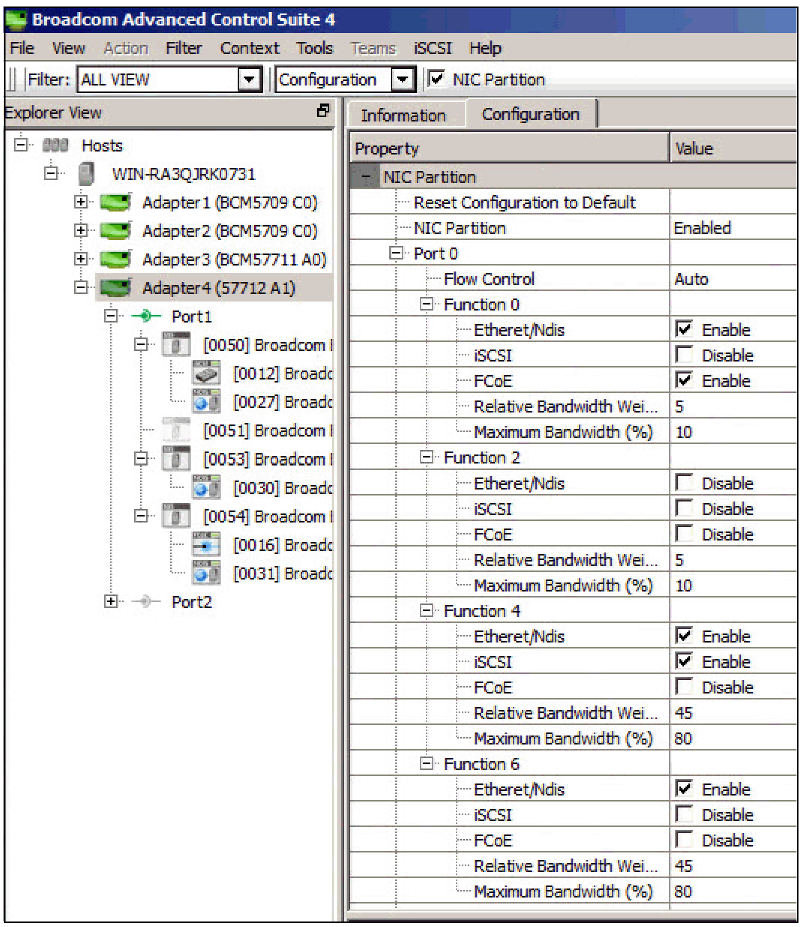
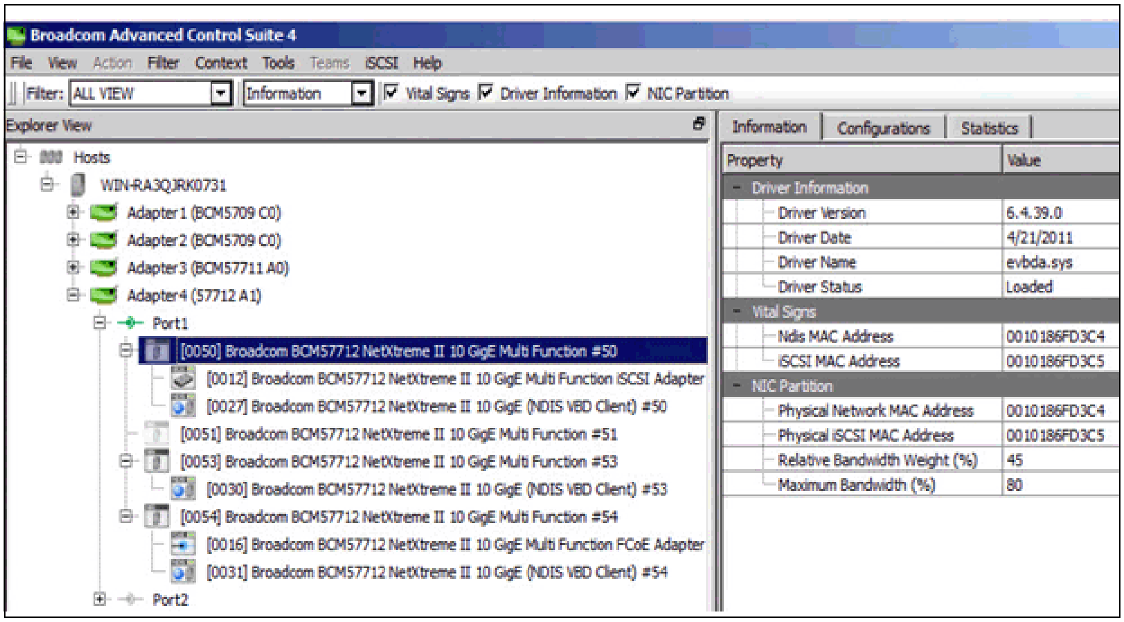
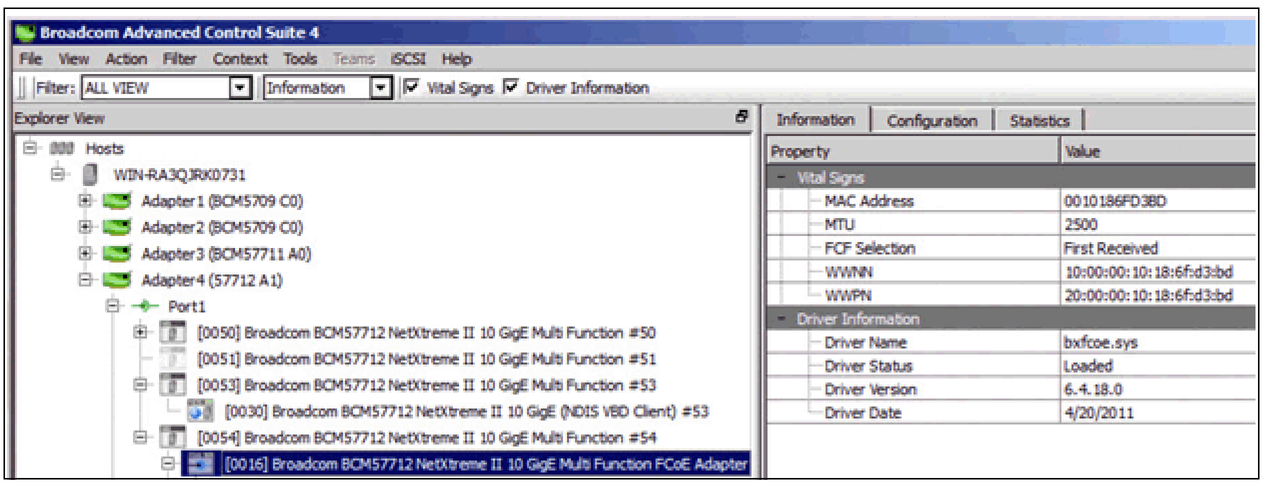


Figure 12: BACS4 Switch Independent Partitioning Settings

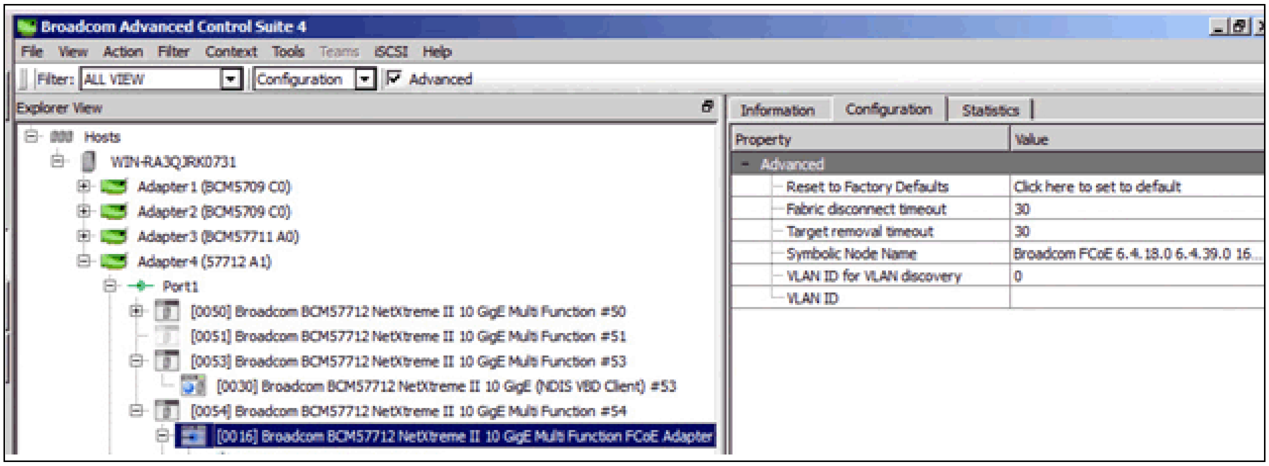
BACS4 displays the per partition Virtual Bus Device (VBD) information.



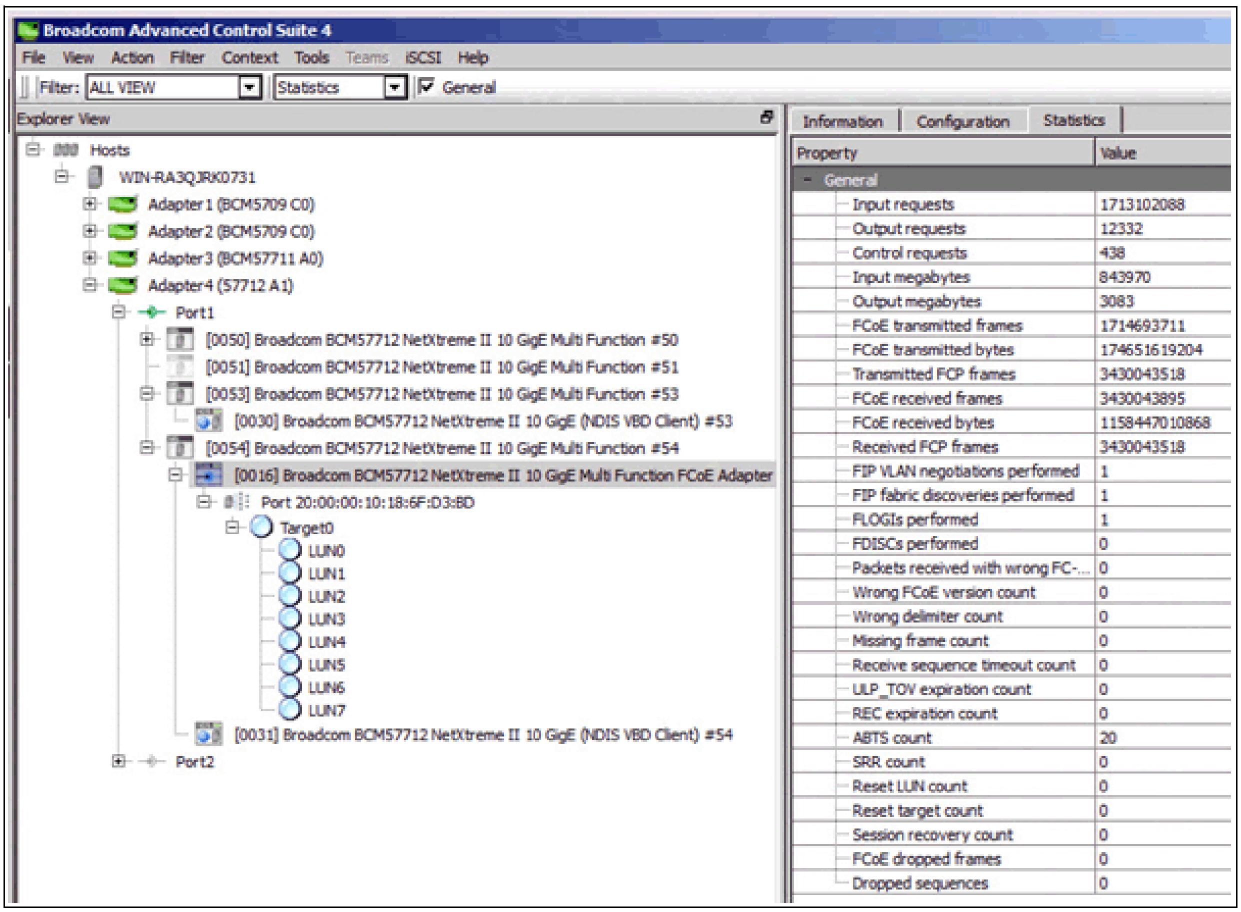
The following shows the per partition FCoE device information.



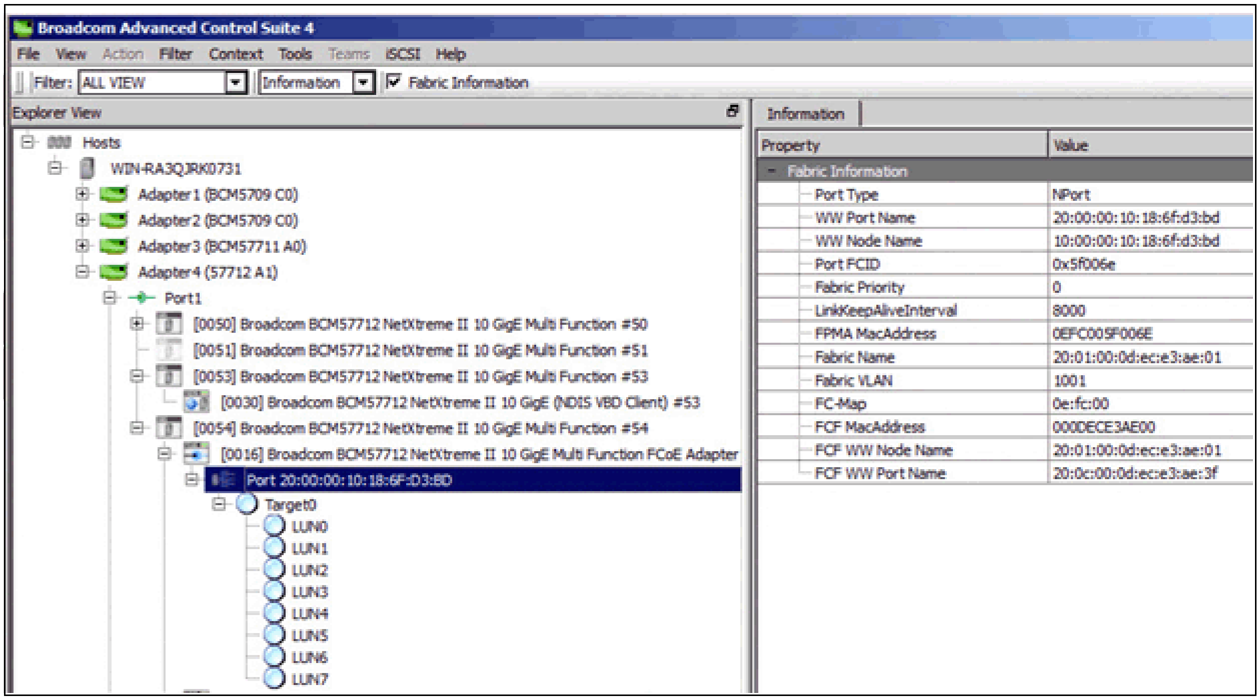
The following shows the configuration of FcoE device settings.



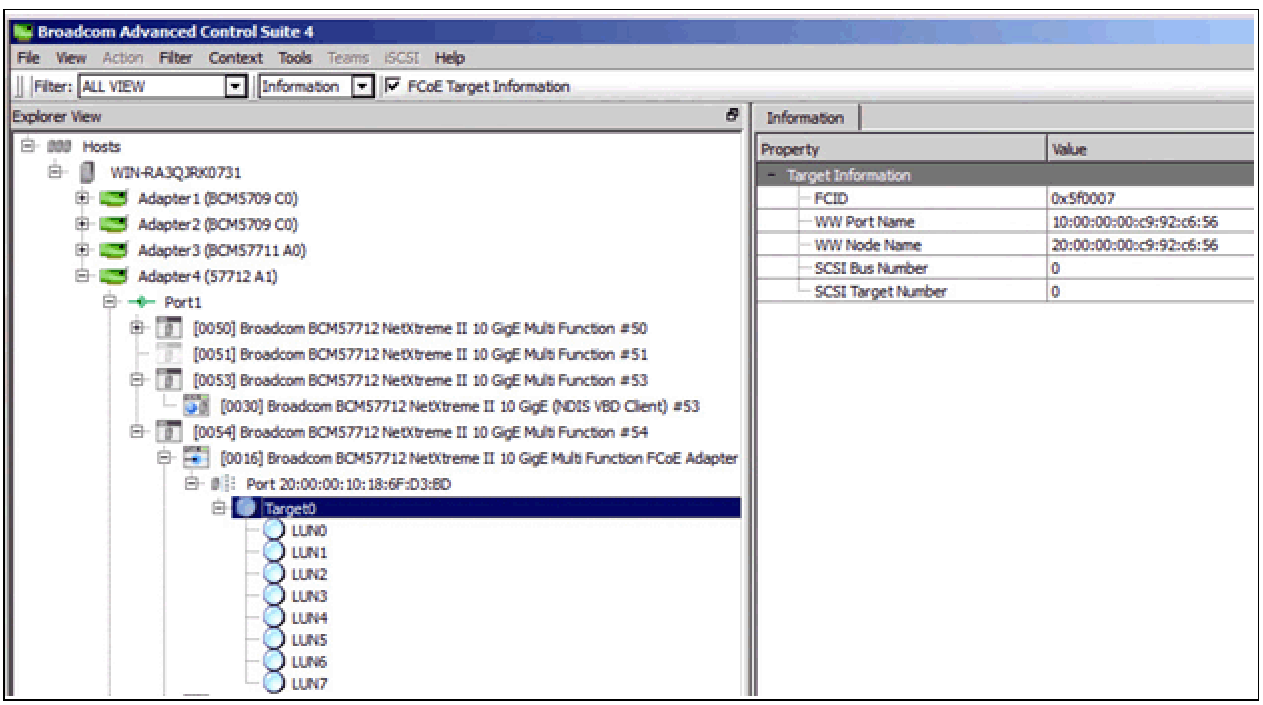
The following shows the per partition FCoE device statistics.



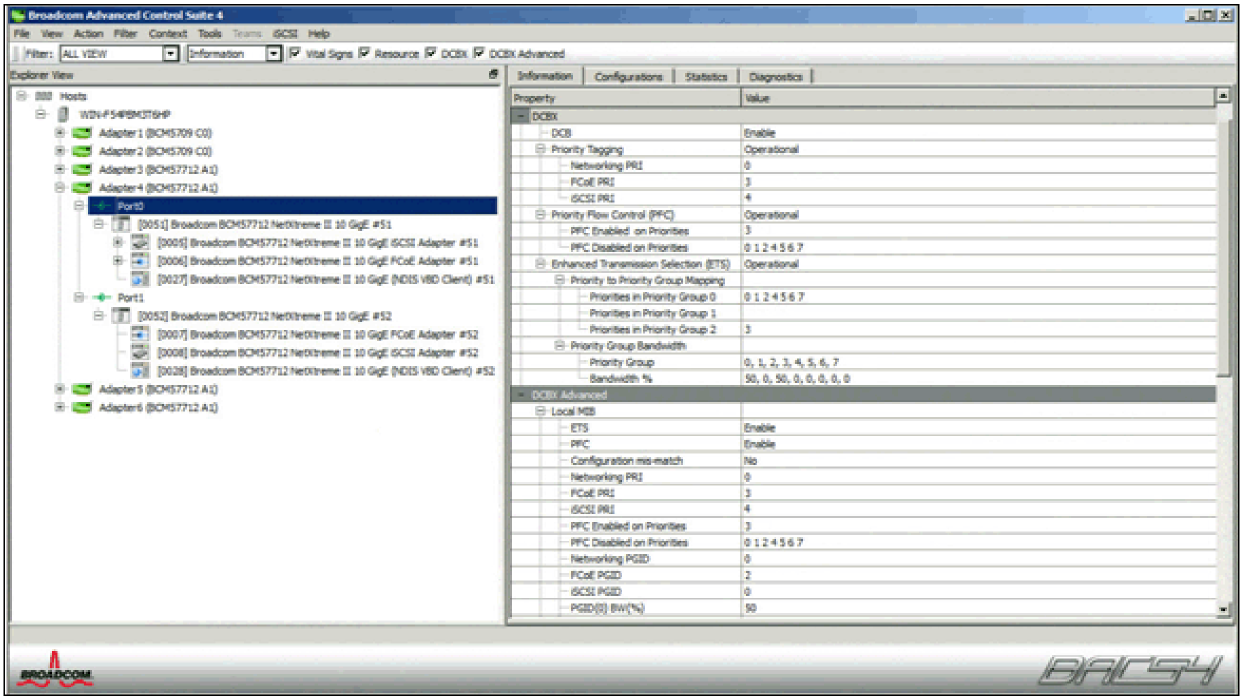
The following shows the per partition FCoE device connection information.



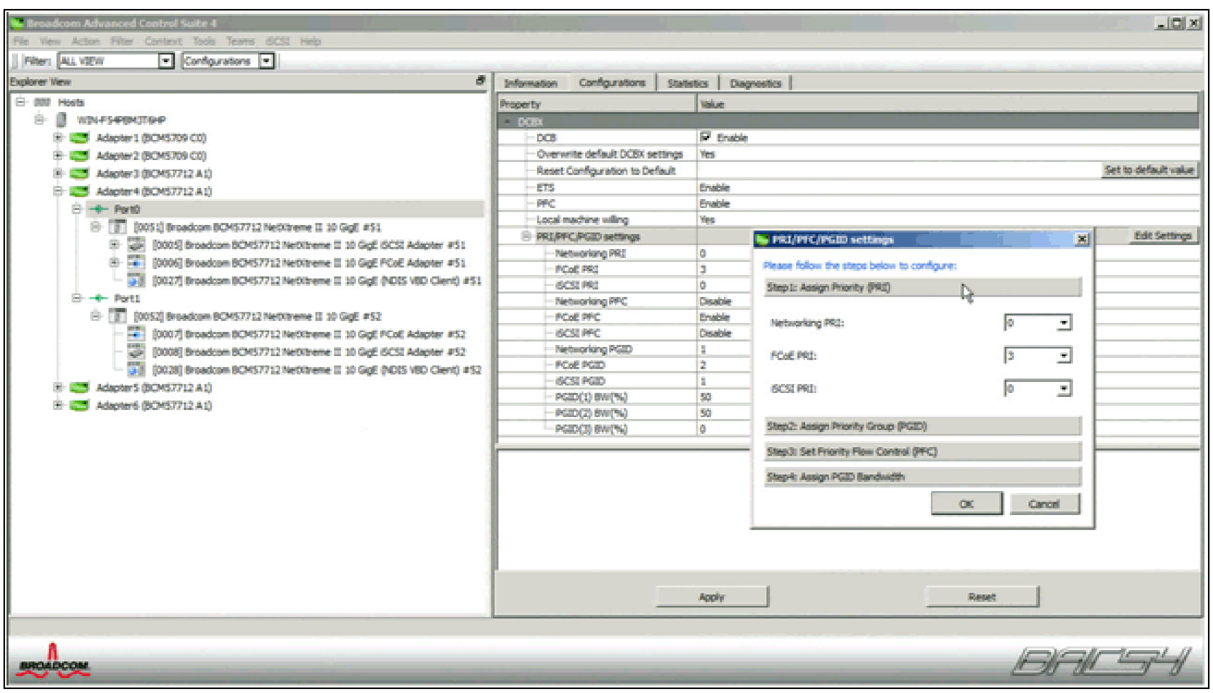
The following shows the per partition per FCoE target information.



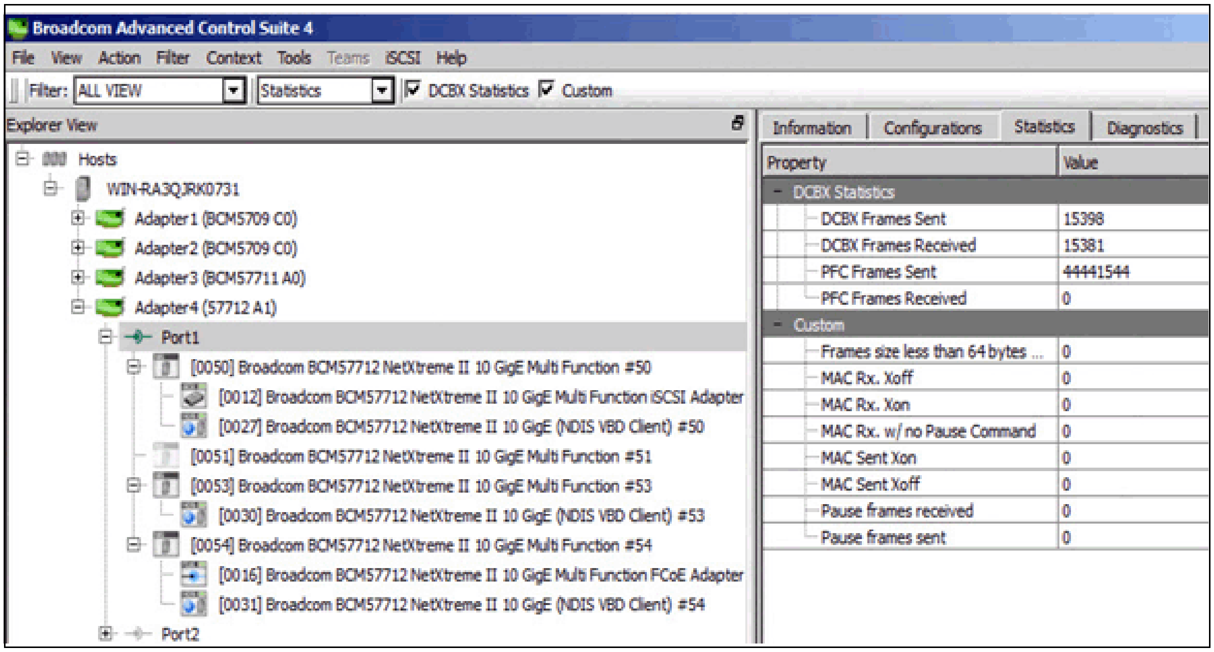
The following shows the per port DCB protocol information



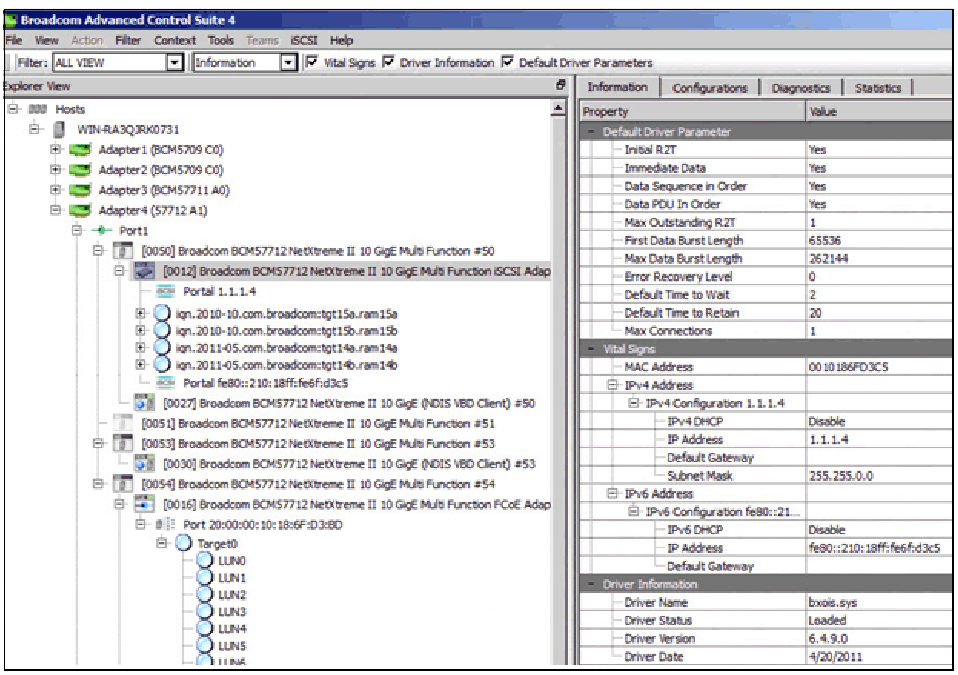
BACS4 allows per port enabling and configuring of the DCB protocol default and initially advertised settings and port “willingness” to change to the received DCBx values as shown below.



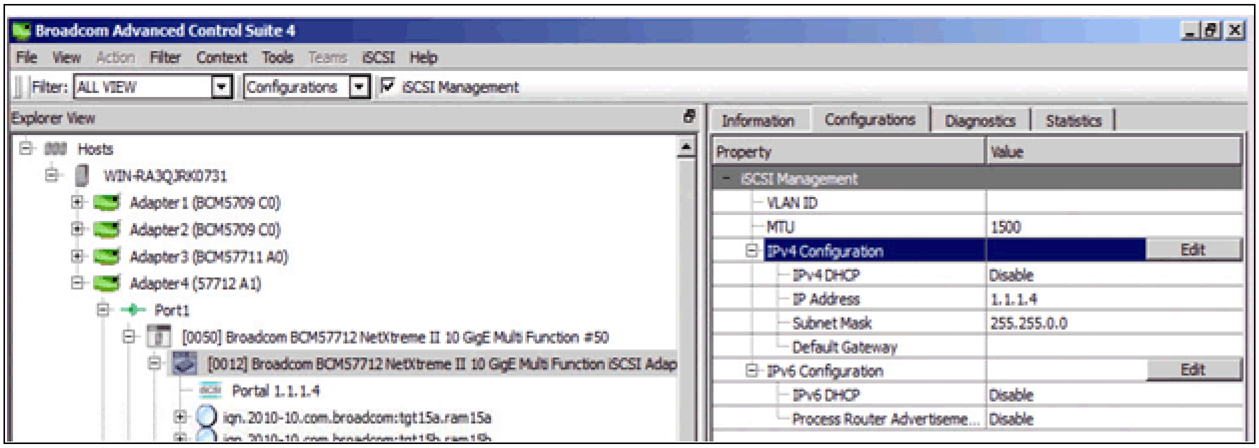
The following shows the per port DCB protocol statistics.



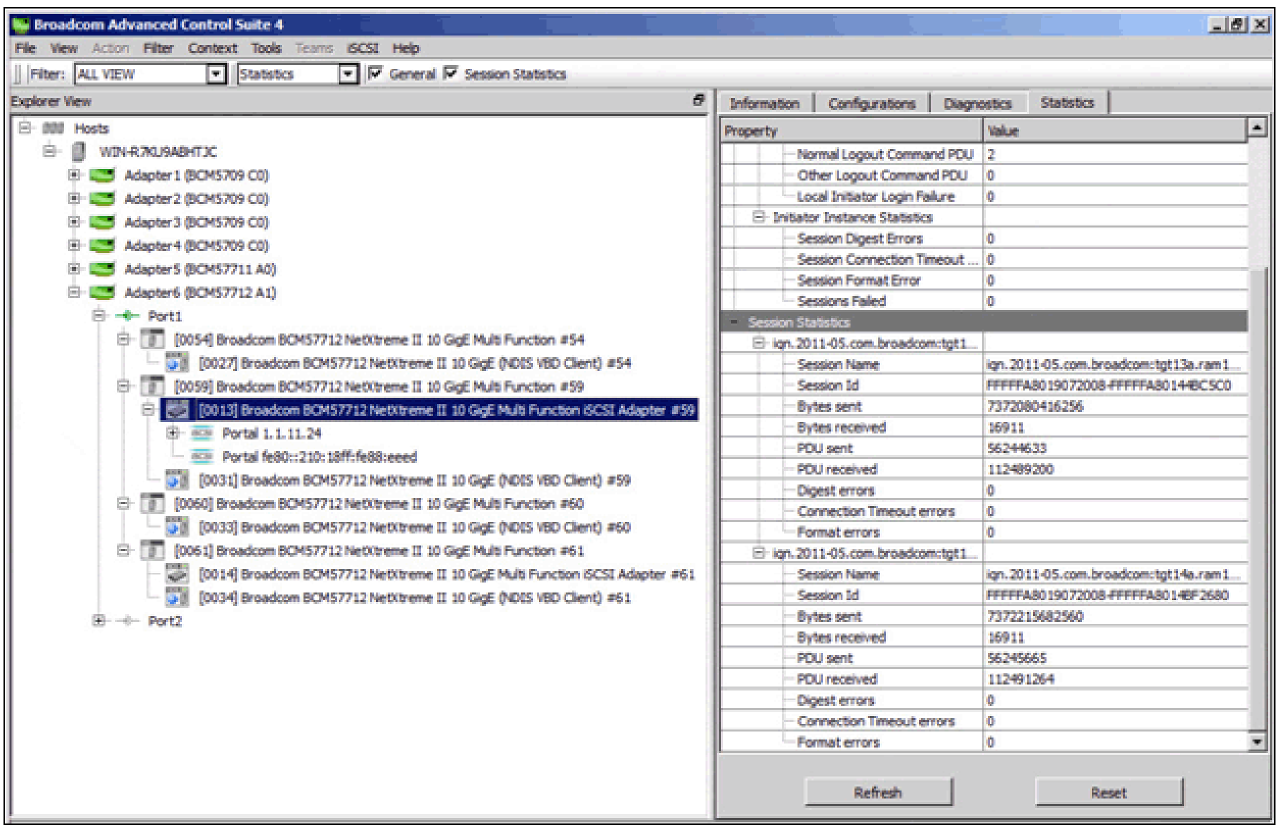
The following shows the per partition iSCSI device information.



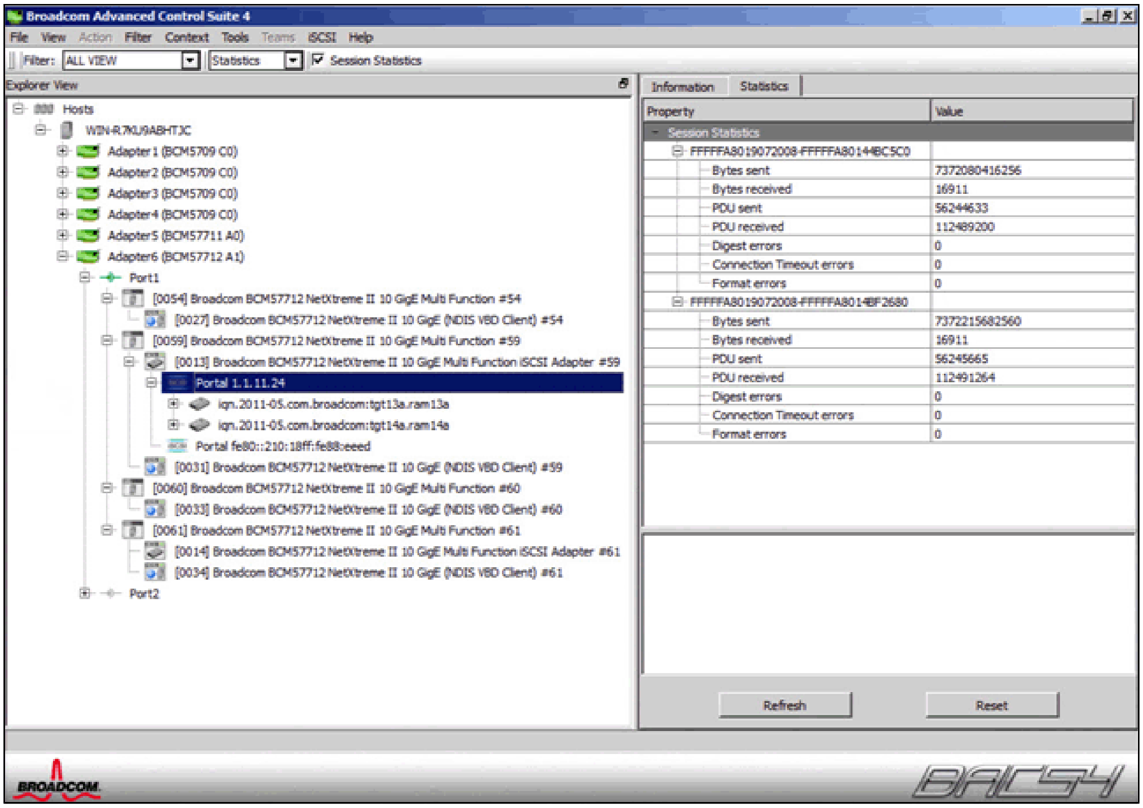
The per partition iSCSI Offload device's VLAN, MTU and IP address (IPv4 and IPv6) settings are configured in BACS4 as shown below.



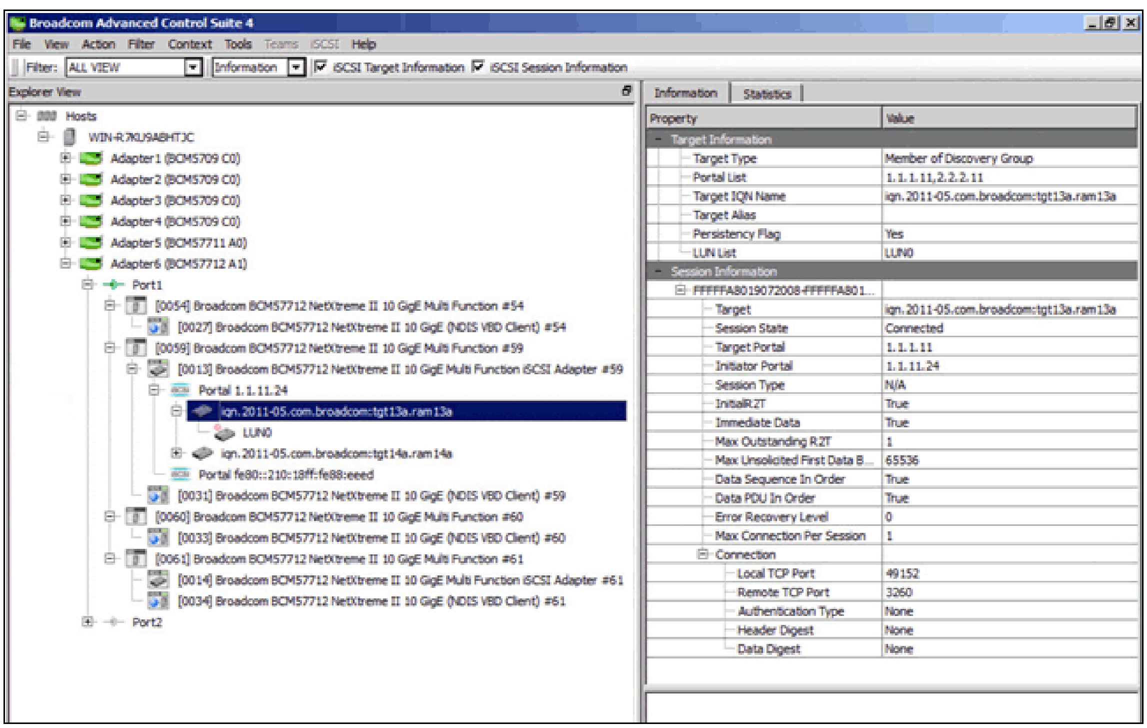
The following shows the per partition iSCSI device traffic statistics.



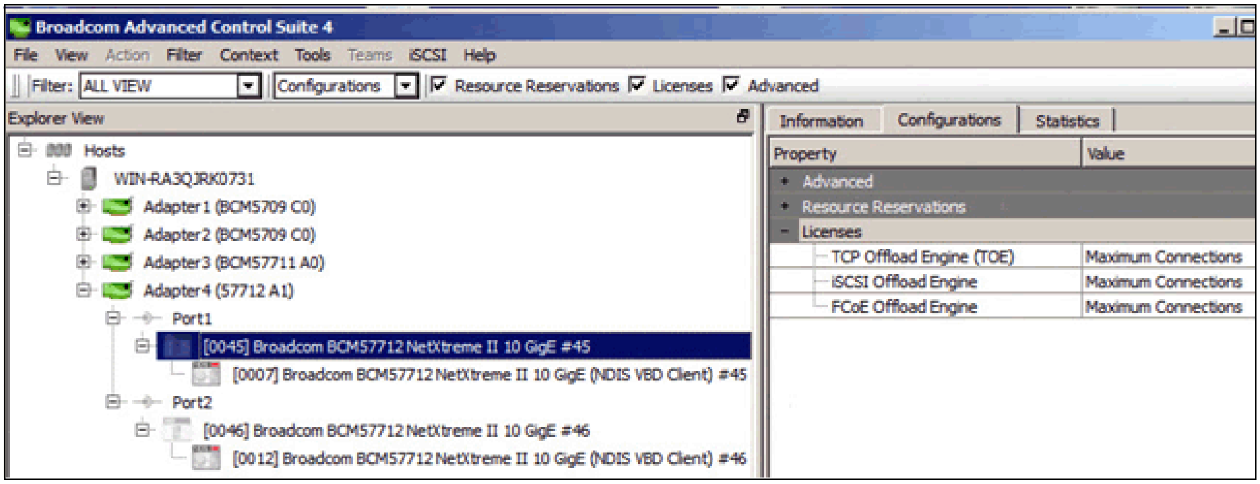
The following shows the per partition iSCSI device per Portal traffic statistics.



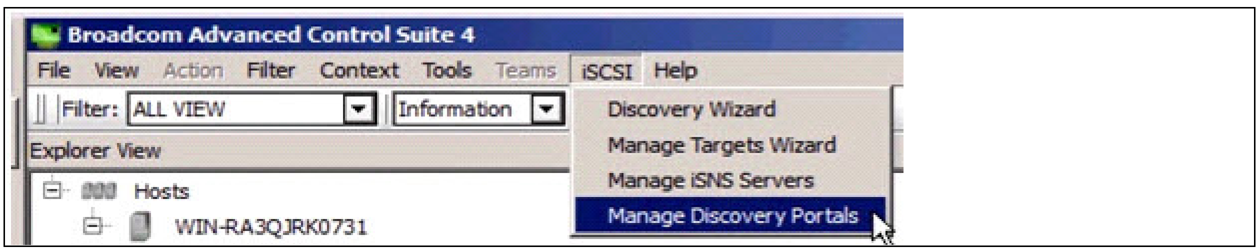
The following shows the per partition iSCSI device per Portal per target information.



The following shows the per device/partition's current offload licenses.

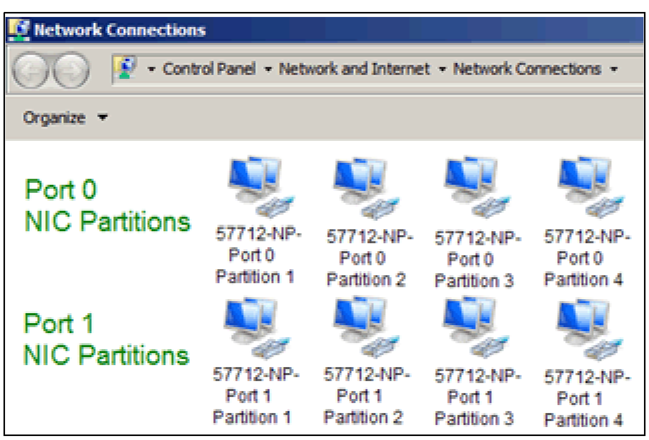


An iSCSI Discovery and Management Wizard is also included with BACS4. This can be used to connect and manage the iSCSI hardware offload enabled initiator devices to various iSCSI targets as shown below.

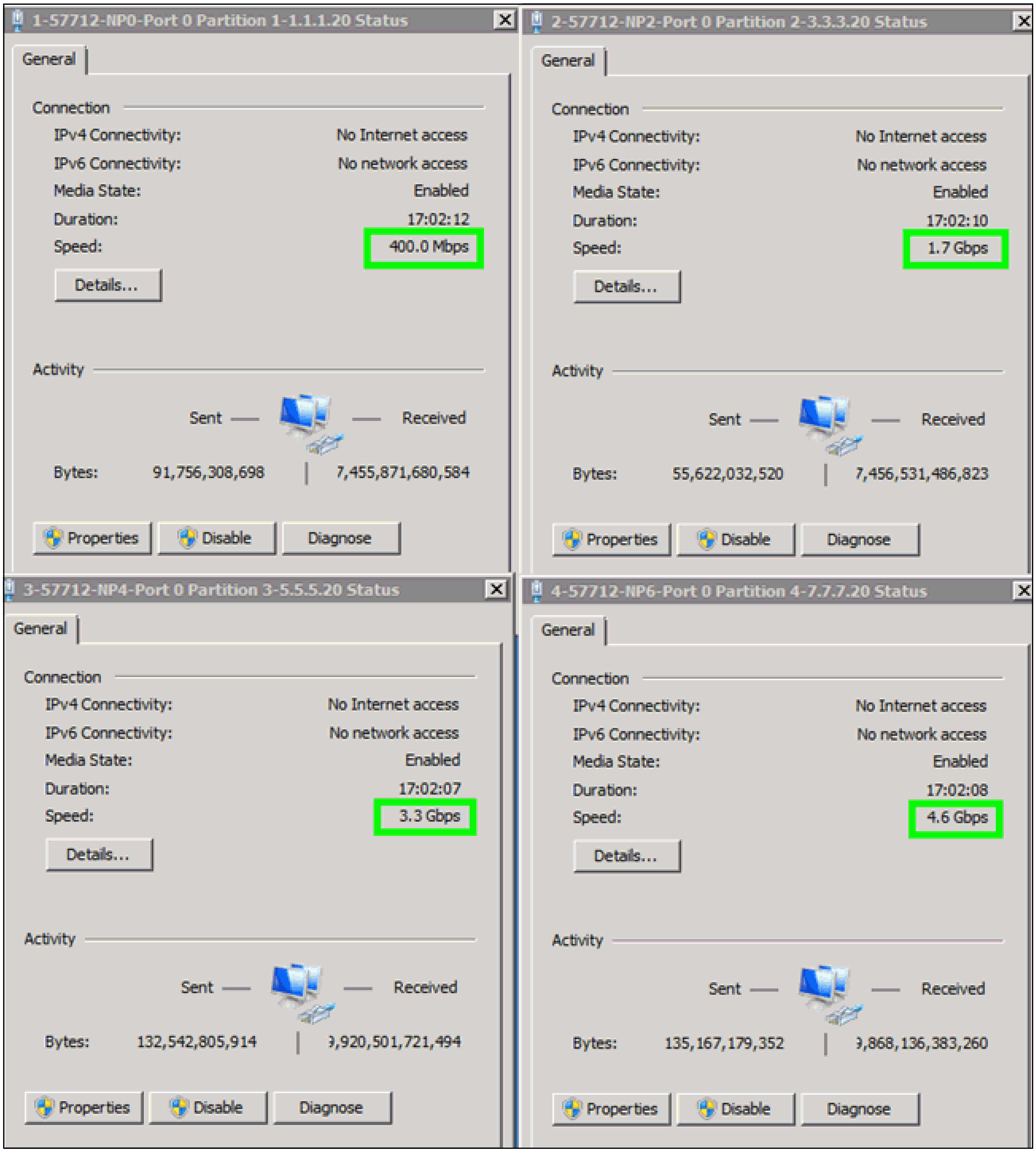


Microsoft Windows Network Connections

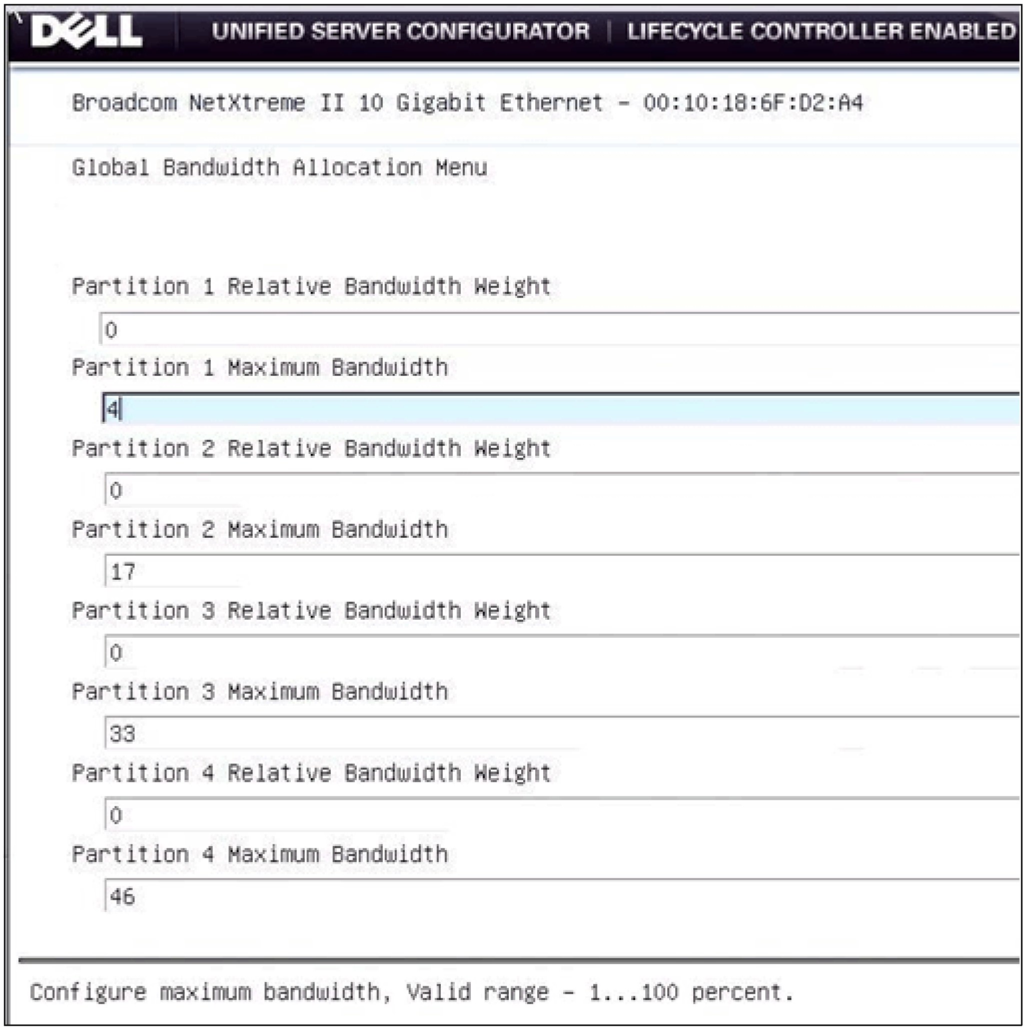
These devices can be used by any application, as if they were a separate adapter port. They appear as separate Ethernet devices in **Network Connections** from the Windows Control Panel. The following shows eight USC/ BACS4-enabled Ethernet Protocol partitions as eight separate Ethernet network connections and these are arranged in port order (0 and 1) and partition order (1 through 4).



Each of these network connection devices can be accessed, individually, as if they where separate adapters. The connection status shows the USC/BACS4 **Maximum Bandwidth** setting as the **Speed** of the connection.

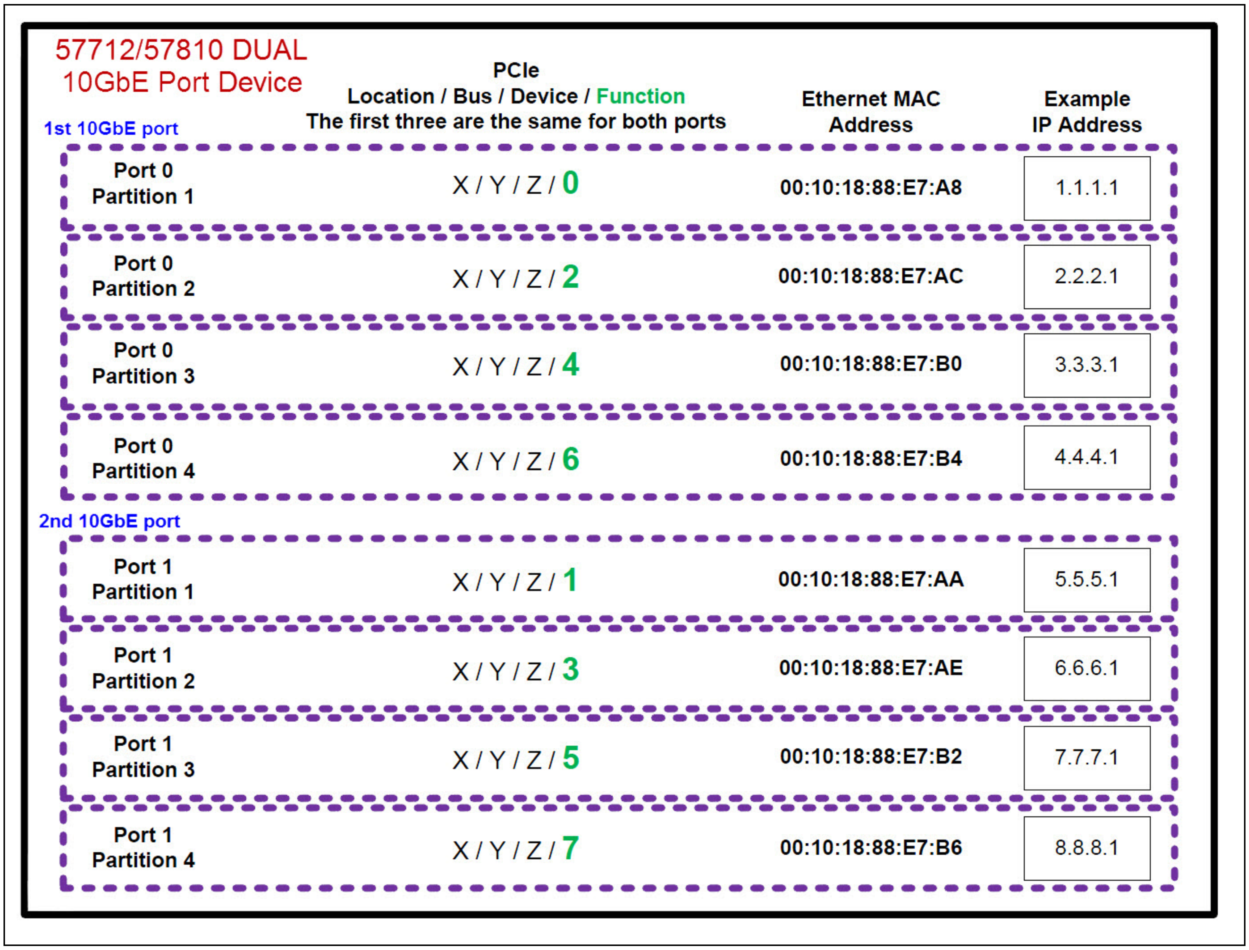


The previous Link Speeds are the result of the following USC **Maximum Bandwidth** settings, and show its (1% of line rate) configurable granularity.



Device PCIe Bus Location

The PCIe interface Location, Bus, and Device position numbers are the same for both ports and all eight of the partitions on those ports. The only PCIe interface location values that are different are the **Function** numbers. In non-partitioned Single Function (SF) mode, you would only have functions 0 and 1. In partitioned (Switch Independent Partitioning) mode, you have functions 0 through 7, with functions 0-2-4-6 existing on the first port and functions 1-3-5-7 existing on the second port. The actual numbering position an adapter is assigned by Windows is not entirely related to the PCIe interface numbering and is more related to what open location position numbers are available in the registry when the adapters get enumerated. Therefore, port 0 partition 1 may not always occupy the first position in the Windows Device Manager's Network Adapters or Storage Controllers or System Devices sections.

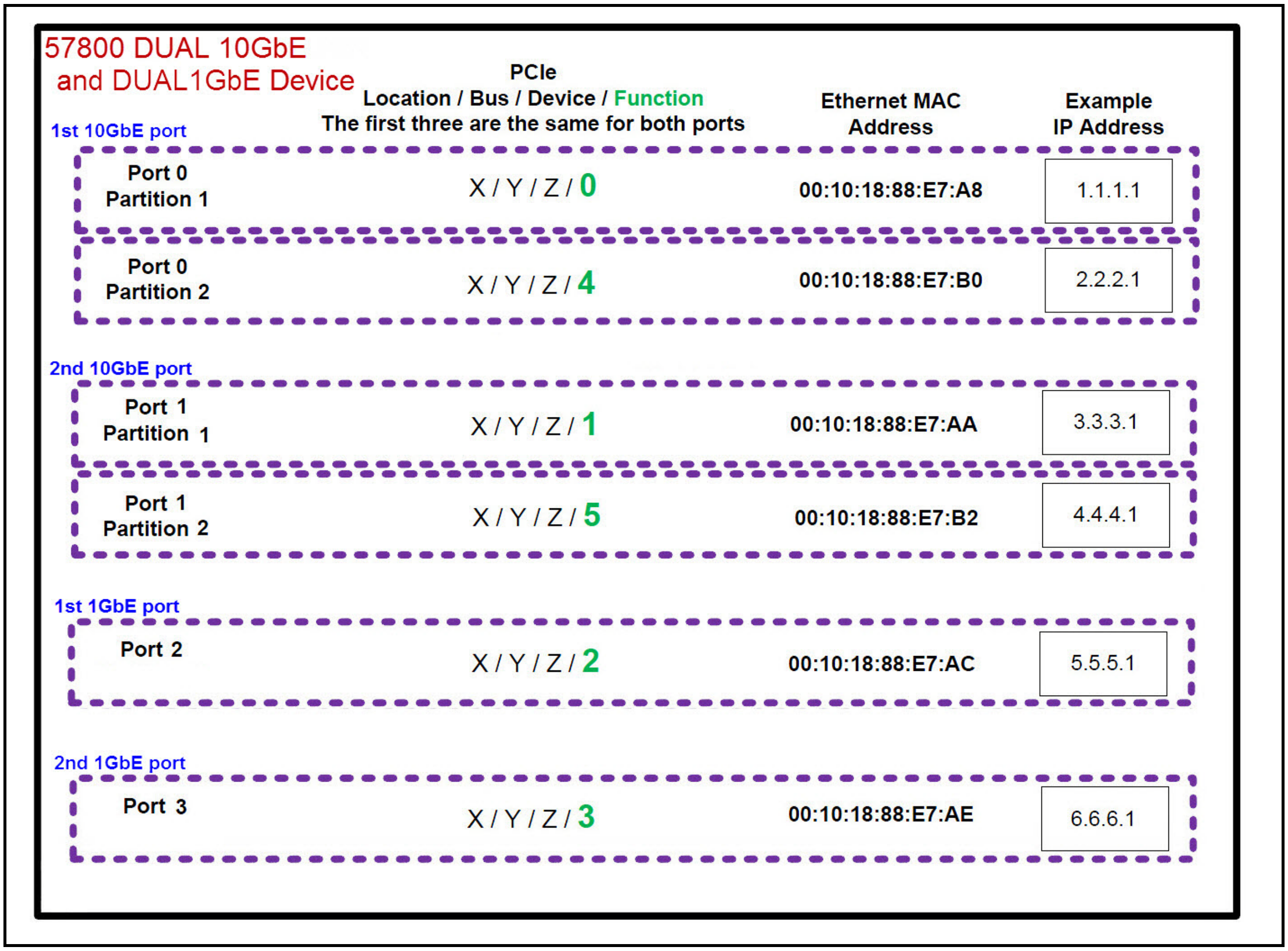


The partition's MAC addresses interleave the two ports, as do the function numbers (see Table 3).

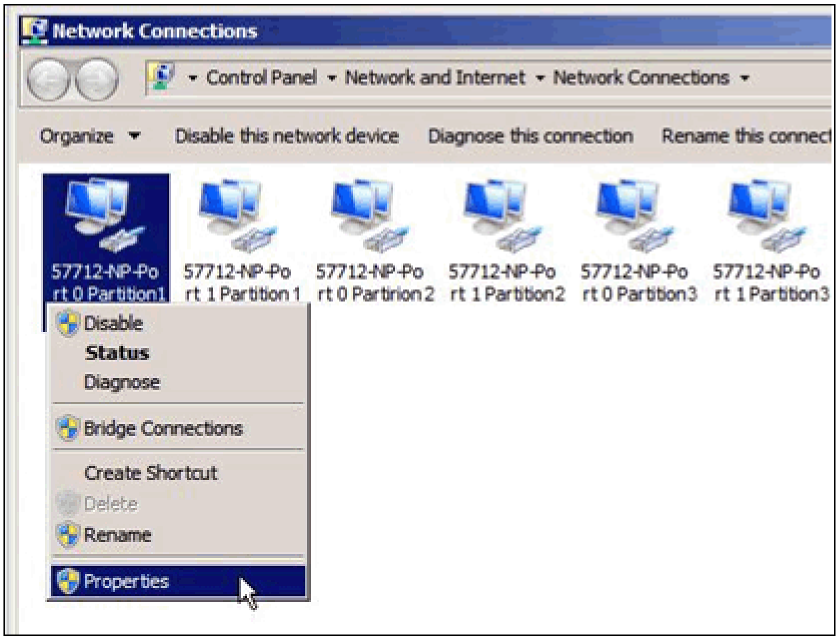
Table 3: Port, Function, MAC Address Example

Port 0, Partition 1	Function 0 = MAC address...:A8
Port 0, Partition 2	Function 2 = MAC address...:AC
Port 0, Partition 3	Function 4 = MAC address...:B0
Port 0, Partition 4	Function 6 = MAC address...:B4
Port 1, Partition 1	Function 1 = MAC address...:AA
Port 1, Partition 2	Function 3 = MAC address...:AE
Port 1, Partition 3	Function 5 = MAC address...:B2
Port 1, Partition 4	Function 7 = MAC address...:B6

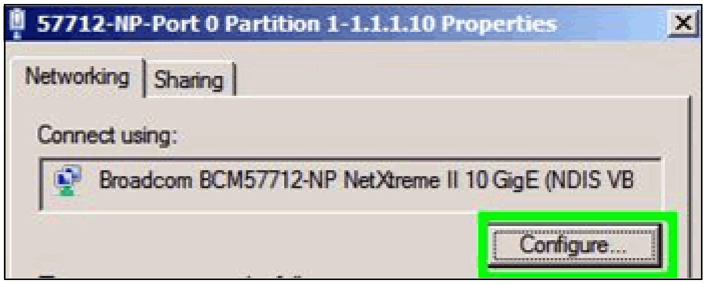
The quad port 57800S (which combines two 10GbE with two 1GbE ports) has slightly different PCI device numbering. The 57800S can have up to two partitions per 10GbE port, where there can be up to two L2 Ethernet protocols plus up to two iSCSI Offload protocols OR one FCoE Offload plus one iSCSI Offload protocols enabled per port. The two dedicated 1GbE ports will remain in single function mode and do not support FCoE or iSCSI Offload modes. The following figure illustrates the 57800S PCI bus numbering.



One way to locate PCIe information is to open the individual Network Connection's Properties.

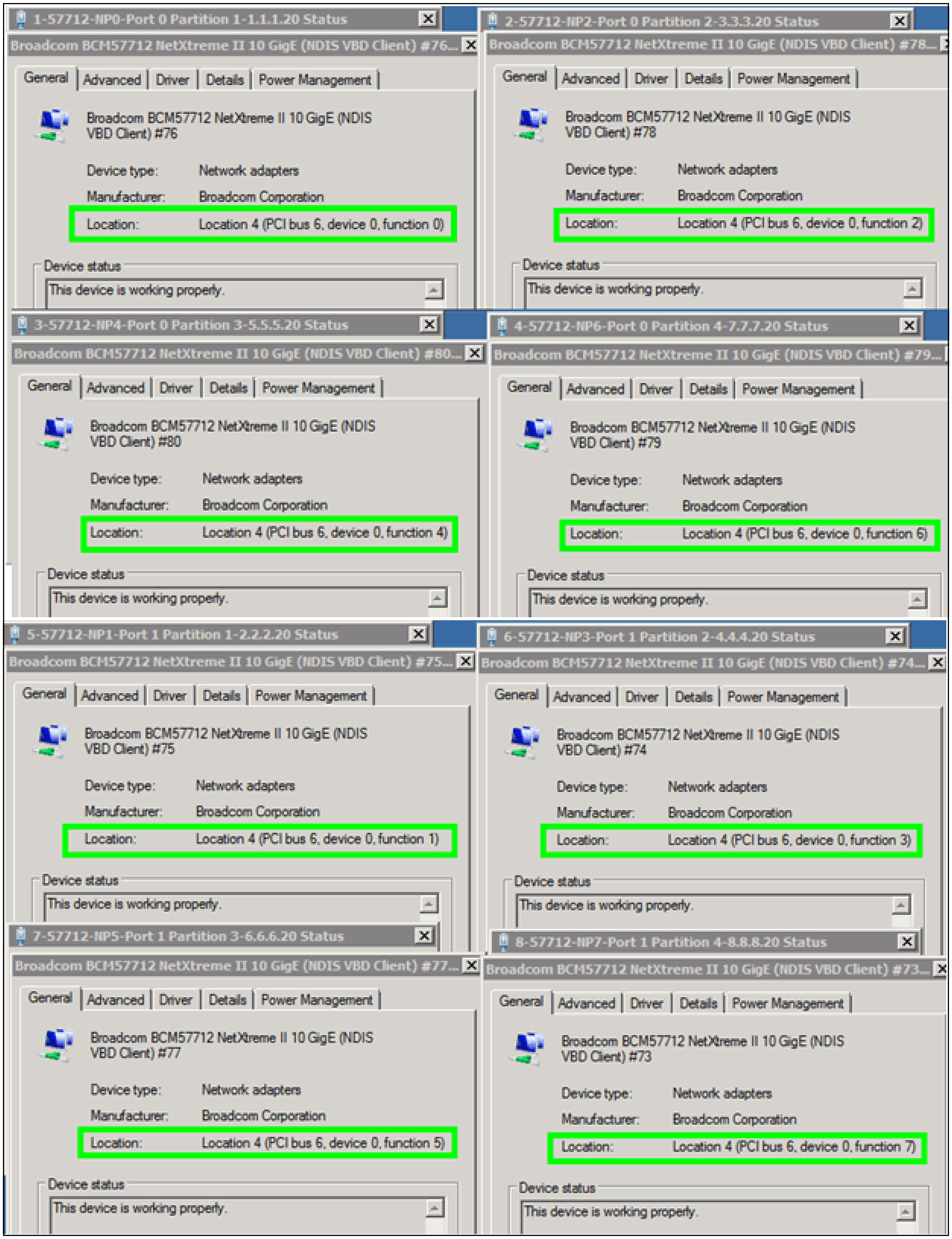

Network Connections
Control Panel
Network and Internet
Network Connections
Organize
Disable this network device
Diagnose this connection
Rename this connect
57712-NP-Port 0 Partition1
57712-NP-Port 1 Partition 1
57712-NP-Port 0 Partirion 2
57712-NP-Port 1 Partition2
57712-NP-Port 0 Partition3
57712-NP-Port 1 Partition3
Disable
Status
Diagnose
Bridge Connections
Create Shortcut
Delete
Rename
Properties

In the Properties window, select the device **Configure** button.



In the NDIS client device properties, you will find that connection's PCIe bus, device, function location information. For partitioned adapters, locate the function number that provides the partition that this connection is connected. The same can be done with Device Manager, especially for iSCSI Storage devices. All of the enabled devices on the same partition have identical PCIe interface location information, with only the function number varying. The following shows the eight Ethernet-partitioned adapter's PCIe device location information.

Notice the Windows enumerated device numbering (**#**) values do not follow the PCIe bus function numbering, nor the port to partition numbering.


1-57712-NP0-Port 0 Partition 1-1.1.1.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #76...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #76
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 0)
Device status
This device is working properly.
2-57712-NP2-Port 0 Partition 2-3.3.3.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #78...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #78
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 2)
Device status
This device is working properly.
3-57712-NP4-Port 0 Partition 3-5.5.5.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #80...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #80
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 4)
Device status
This device is working properly.
4-57712-NP6-Port 0 Partition 4-7.7.7.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #79...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #79
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 6)
Device status
This device is working properly.
5-57712-NP1-Port 1 Partition 1-2.2.2.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #75...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #75
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 1)
Device status
This device is working properly.
6-57712-NP3-Port 1 Partition 2-4.4.4.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #74...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #74
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 3)
Device status
This device is working properly.
7-57712-NP5-Port 1 Partition 3-6.6.6.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #77...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #77
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 5)
Device status
This device is working properly.
8-57712-NP7-Port 1 Partition 4-8.8.8.20 Status
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #73...
General Advanced Driver Details Power Management
Broadcom BCM57712 NetXtreme II 10 GigE (NDIS VBD Client) #73
Device type: Network adapters
Manufacturer: Broadcom Corporation
Location: Location 4 (PCI bus 6, device 0, function 7)
Device status
This device is working properly.

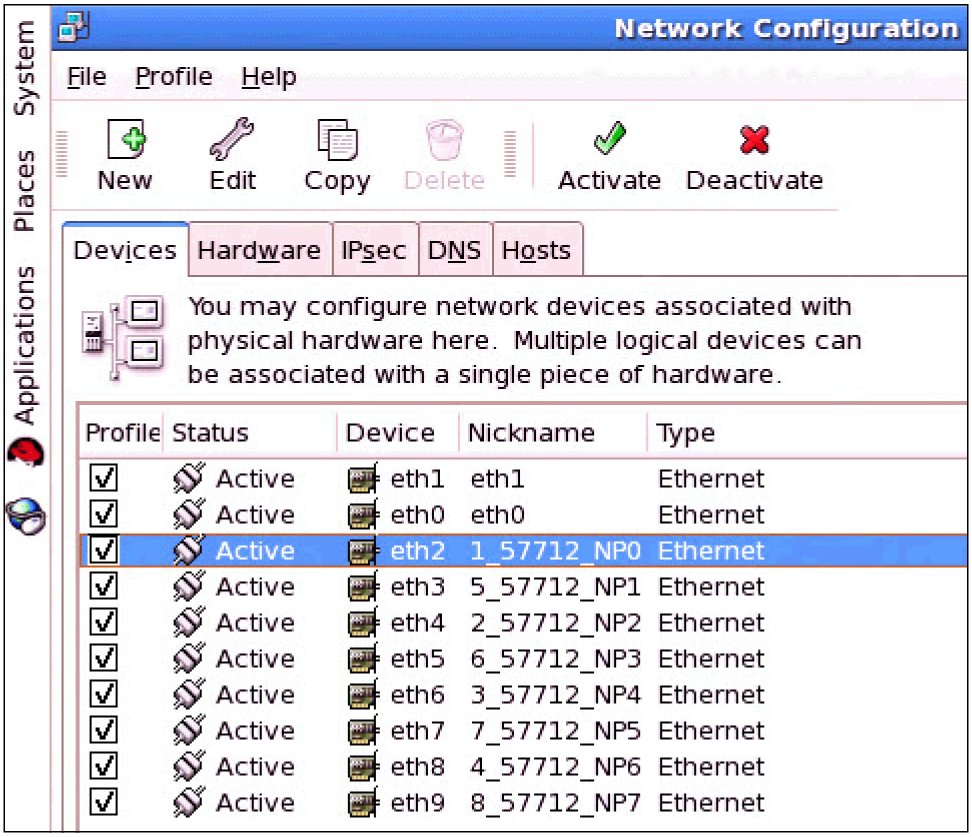
Red Hat Enterprise Linux v6.x

Linux shows the respective device protocols that were enabled in USC if the NetXtreme II device drivers are installed. Go to the Dell Driver Download web site for the latest Linux device drivers (software version 17.2.x or later and firmware version 7.2.x or later) and ensure they are installed on your system. In Linux, the Ethernet Protocol is always enabled on all eight partitions, and FCoE Offload (HBA) Protocol can be enabled on any single partition of each port and the iSCSI Offload (HBA) Protocol can be enabled on any remaining single partition of that port OR, if the FCoE protocol is not enabled, then on any two partitions of each port.

In newer versions of Linux, the "ethtool -s ethX speed NNNNN" command supports setting the 57712-k/578xxS partition’s Maximum Bandwidth value without a system reboot, where ethX is the partition and NNNNN is the speed which is in 1 Mbps increments (i.e., it is NOT a percentage):

- For a 10GbE link connection, the range is from 100 (which is equivalent to 100 Mbps or 1% of the 10GbE link speed) to 10000 (which is equivalent to 10 Gbps or 100% of the 10GbE link speed)—the last two zero positions are not used.
- For a 1GbE link connection, the range is from 10 (which is equivalent to 10 Mbps or 1% of the 1GbE link speed) to 1000 (which is equivalent to 1 Gbps or 100% of the 1GbE link speed)—the last zero position is not used.

The following shows the RHEL Network Configuration page with the eight enabled Ethernet protocol partitions (always four per port) as the Broadcom 57712-k/578xxS NetXtreme II 10 GigE Ethernet devices Eth2 through Eth9 (in this example). Linux enumerates the partitions in order of the PCI function numbers, which is slightly different from Windows where port 0 has functions/partitions 0/2/4/6 (which are eth 2/4/6/8) and port 1 has functions/partitions 1/3/5/7 (which are eth 3/5/7/9).



System
Places
Applications
Network Configuration
File
Profile
Help
New
Edit
Copy
Delete
Activate
Deactivate
Devices
Hardware
IPsec
DNS
Hosts
You may configure network devices associated with physical hardware here. Multiple logical devices can be associated with a single piece of hardware.
Profile Status Device Nickname Type
Active eth1 eth1 Ethernet
Active eth0 eth0 Ethernet
Active eth2 1_57712_NP0 Ethernet
Active eth3 5_57712_NP1 Ethernet
Active eth4 2_57712_NP2 Ethernet
Active eth5 6_57712_NP3 Ethernet
Active eth6 3_57712_NP4 Ethernet
Active eth7 7_57712_NP5 Ethernet
Active eth8 4_57712_NP6 Ethernet
Active eth9 8_57712_NP7 Ethernet

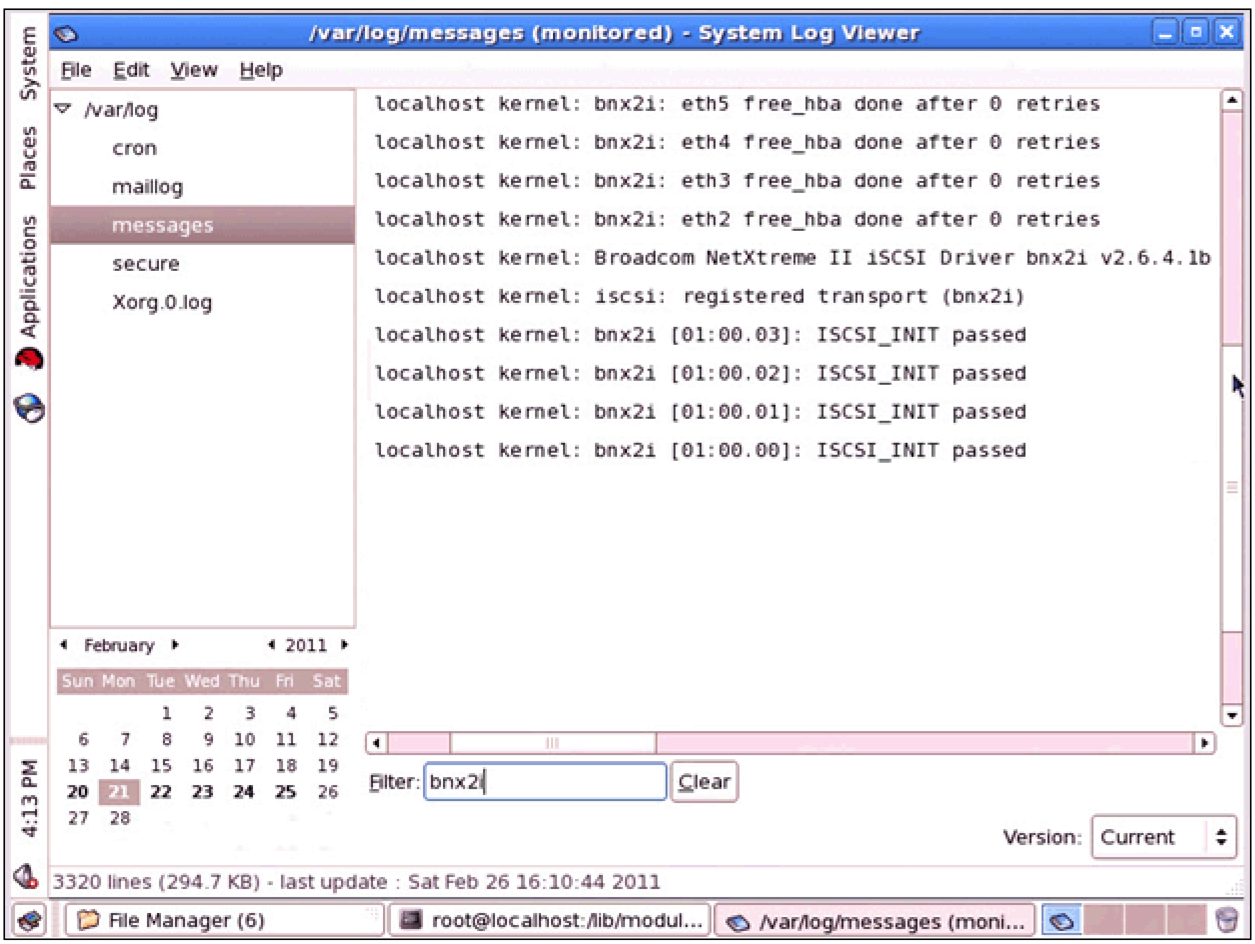
Linux's `ifconfig` command shows the partition's eight Ethernet protocol devices and various statistics.

```
eth2      Link encap:Ethernet  HWaddr 00:10:18:88:E7:A8
          inet addr:10.1.1.200  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7a8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:27937 errors:0 dropped:0 overruns:0 frame:0
          TX packets:40 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5030387 (4.7 MiB)  TX bytes:9923 (9.6 KiB)
          Interrupt:169 Memory:d1800000-d1ffffff

eth3      Link encap:Ethernet  HWaddr 00:10:18:88:E7:AA
          inet addr:10.1.1.201  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7aa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:27500 errors:0 dropped:0 overruns:0 frame:0
          TX packets:36 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4946656 (4.7 MiB)  TX bytes:9456 (9.2 KiB)
          Interrupt:225 Memory:d2800000-d2ffffff

eth4      Link encap:Ethernet  HWaddr 00:10:18:88:E7:AC
          inet addr:10.1.1.202  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7ac/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:27034 errors:0 dropped:0 overruns:0 frame:0
          TX packets:42 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4860331 (4.6 MiB)  TX bytes:10027 (9.7 KiB)
          Interrupt:225 Memory:d3800000-d3ffffff
```

```
th5       Link encap:Ethernet  HWaddr 00:10:18:88:E7:AE
          inet addr:10.1.1.203  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7ae/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:26782 errors:0 dropped:0 overruns:0 frame:0
          TX packets:34 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4807217 (4.5 MiB)  TX bytes:8866 (8.6 KiB)
          Interrupt:204 Memory:d4800000-d4ffffff

th6       Link encap:Ethernet  HWaddr 00:10:18:88:E7:B0
          inet addr:10.2.2.200  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:2682 errors:0 dropped:0 overruns:0 frame:0
          TX packets:34 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:472737 (461.6 KiB)  TX bytes:9139 (8.9 KiB)
          Interrupt:204 Memory:d5800000-d5ffffff

th7       Link encap:Ethernet  HWaddr 00:10:18:88:E7:B2
          inet addr:10.2.2.201  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:2233 errors:0 dropped:0 overruns:0 frame:0
          TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:393371 (384.1 KiB)  TX bytes:8381 (8.1 KiB)
          Interrupt:181 Memory:d6800000-d6ffffff
```

```
eth8      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B4
          inet addr:10.2.2.202  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:1701 errors:0 dropped:0 overruns:0 frame:0
          TX packets:31 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:290494 (283.6 KiB)  TX bytes:8110 (7.9 KiB)
          Interrupt:181 Memory:d7800000-d7ffffff

eth9      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B6
          inet addr:10.2.2.203  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:1730 errors:0 dropped:0 overruns:0 frame:0
          TX packets:33 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:299886 (292.8 KiB)  TX bytes:8450 (8.2 KiB)
          Interrupt:169 Memory:d8800000-d8ffffff
```

Check the Linux system message logs to see if the iSCSI HBAs (in this example) on the first two partitions of each port (Port 0, Partition 1 = eth2; Port 0, Partition 2 = eth4; Port 1, Partition 1 = eth3; and Port 1 Partition 2 = eth5) have iSCSI Offload enabled.



The Linux System Log Viewer message for the FCoE initiator being detected on scsi5 might look like this:

```
...  kernel: scsi5 : Broadcom Offload FCoE Initiator
```

The Linux System Log Viewer message for the FCoE driver (bnx2fc) being loaded on scsi6 might look like this:

```
...  kernel: bnx2fc: Broadcom NetXtreme II FCoE Driver bnx2fc v2.2.9 (Feb 28, 2012)
...  kernel: bnx2fc: Entered bnx2fc_ulp_init
...  kernel: bnx2fc: min xid 0x0, max xid 0xfff
...  kernel: bnx2fc: [06]: FCOE_INIT passed
...  kernel: bnx2fc: bnx2fc started.
```

For iSCSI, also check the **iscsi_host** folder to see if your iSCSI devices are present.

```
[root@localhost]# cd /sys/class/iscsi_host/

[root@localhost iscsi_host]# pwd

/sys/class/iscsi_host

[root@localhost iscsi_host]# ll

total 0

lrwxrwxrwx 1 root root 0 Jun 29 11:15 host3 -> ../../devices/pci0000:00/0000:00:09.0/0000:07:00.1/
host3/iscsi_host/host3

lrwxrwxrwx 1 root root 0 Jun 29 11:15 host4 -> ../../devices/pci0000:00/0000:00:09.0/0000:07:00.0/
host4/iscsi_host/host4
```

Each installed iSCSI device will appear here. The iSCSI devices uses the **bnx2i** driver which can be checked for with the `lsmod | grep bnx2i` command.

Additionally, check the **fc_host** folder to see if your FCoE devices are present.

```
[root@localhost]# cd /sys/class/fc_host/

[root@localhost fc_host]# pwd

/sys/class/fc_host

[root@localhost fc_host]# ll

total 0

lrwxrwxrwx 1 root root 0 Jun 29 11:11 host8 -> ../../devices/pci0000:00/0000:00:07.0/0000:05:00.0/
host5/fc_host/host5

lrwxrwxrwx 1 root root 0 Jun 29 11:11 host8 -> ../../devices/pci0000:00/0000:00:07.0/0000:06:00.0/
host6/fc_host/host6
```

Each installed FCoE device will appear here. FCoE uses the **bnx2fc** driver which can be checked for with the `lsmod | grep bnx2fc` command.

Another useful command is `sg_map -i -x` which will show all SCSI LUN devices visible to the host.

For Fiber Channel, another useful application is `FCInfo` utility that displays the FCoE HBA port information. FCInfo is also available on Solaris and Windows OS's.

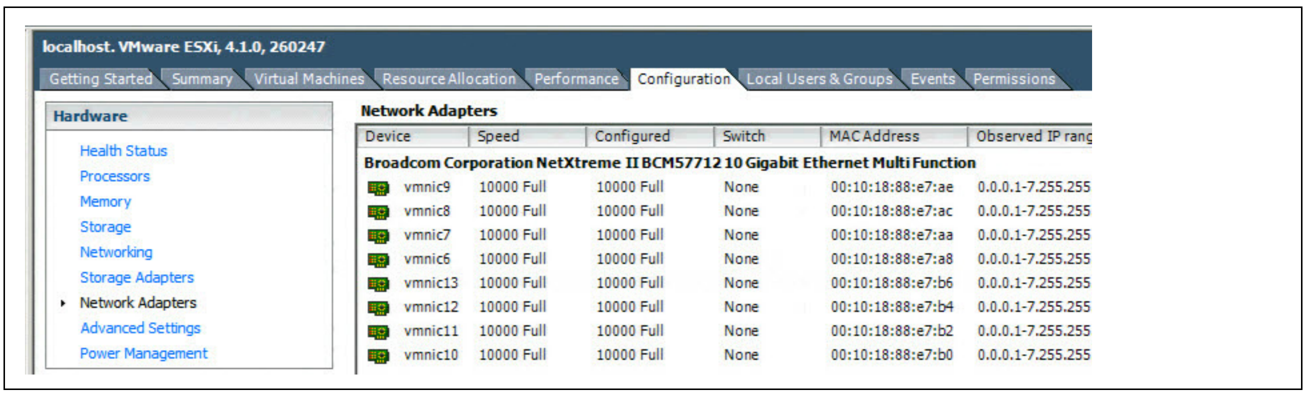
Another useful tool for FCoE is the FCoE Administration Utility (fcoeadm). In order to use this tool, the user must install, on the host, the FCoE-Utils User Space Package from the Open-FCoE Project website (http://www.open-fcoe.org/git/?p=fcoe/fcoe-utils.git;a=summary).

VMWare ESX/ESXi 4.1

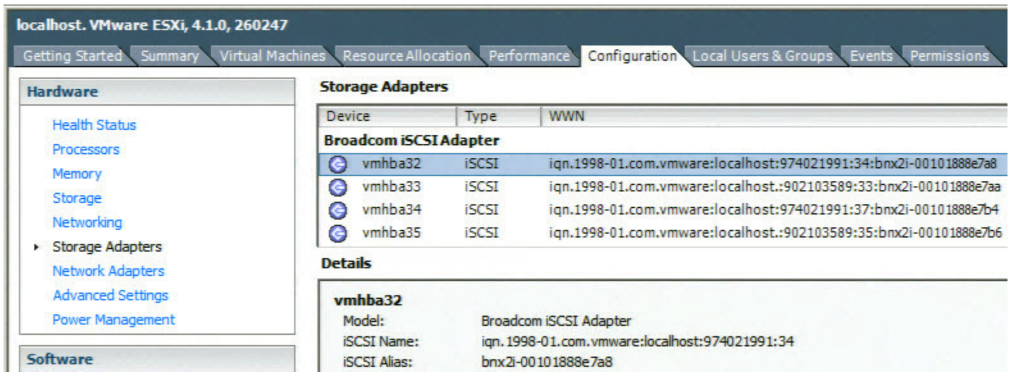
VMWare ESX/ESXi 4.1 shows the respective device protocols that were enabled in USC if the NetXtreme II device drivers are installed. Go to the Dell Driver Download web site (http://support/del.com) for the latest device Firmware - if not already installed. Go to the VMWare web site (http://downloads.vmware.com) for the latest device drivers and insure they are installed on your system. In VMWare, the Ethernet Protocol is always enabled on all eight partitions. VMWare ESX/ESXi 4.1 does support the iSCSI Offload Protocol in both SF and Switch Independent Partitioning mode but does not support the FCoE Offload Protocol in SF or Switch Independent Partitioning modes in VMWare ESX/ESXi 4.1 drivers 1.70.34.v41.1 (bnx2x), 2.70.1k.v41.2 (bnx2i) and Dell firmware version 7.0.47 or later.

Note: VMWare ESX/ESXi 4.1 only supports a maximum of four 10GbE ports (see http://www.vmware.com/pdf/vsphere4/r41/vsp_41_config_max.pdf). Using the 57712-k/57810S Switch Independent Partitioning mode allows you to expand the number of ports usable from 4 physical ports to 16 virtual ports. This allows better port flexibility, traffic isolation, service quality, and bandwidth tuning for your management/backup/migration/production networks.

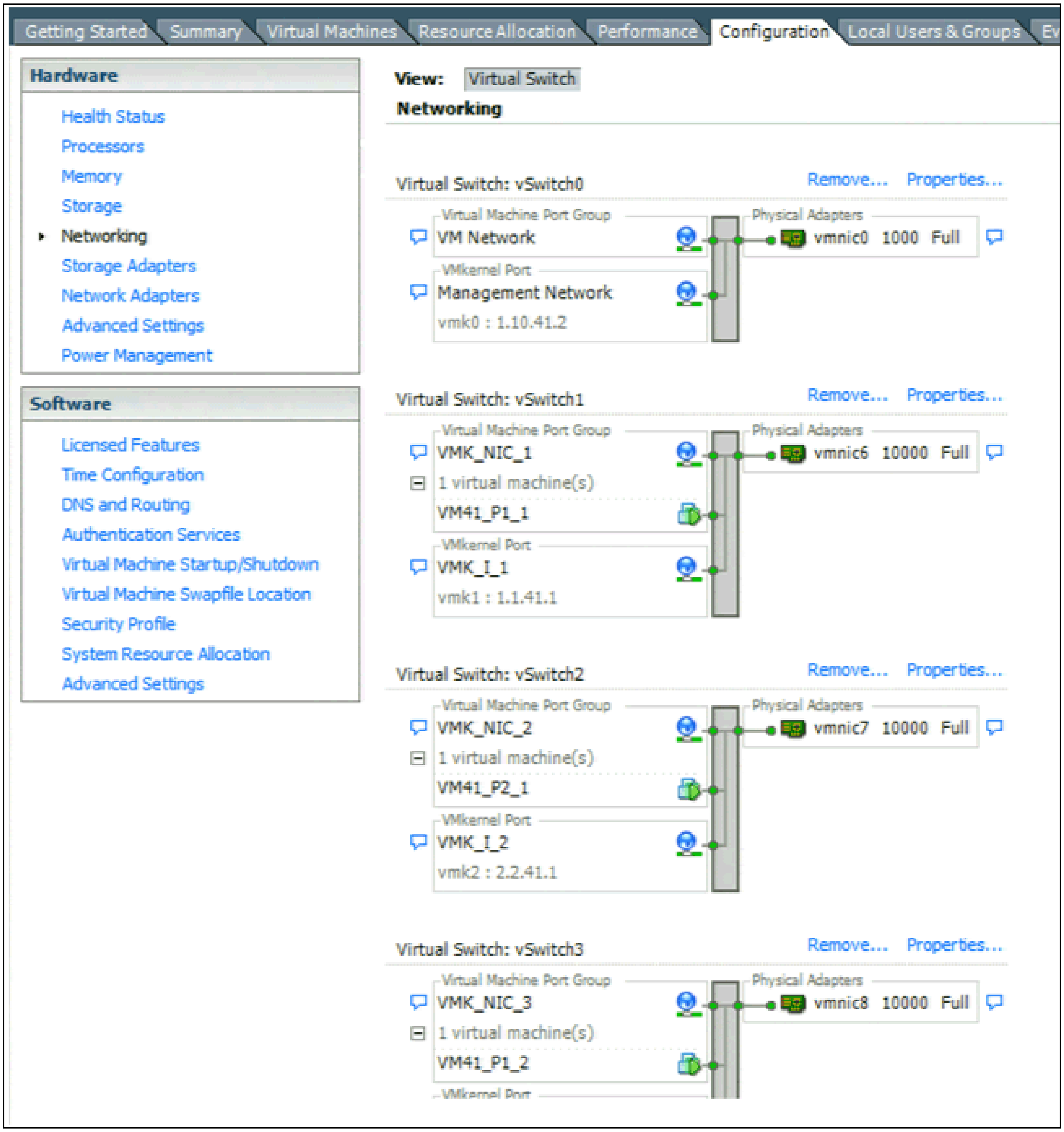
The following shows the vSphere Network Adapters Configuration page with the eight enabled Ethernet protocol partitions (always four per port) as the Broadcom Corporation NetXtreme II BCM57712 10 Gigabit Ethernet MultiFunction devices vmnic6 through vmnic13 (in this example). VMWare enumerates the partitions in the order of the PCI function numbers where port 0 has device functions 0/2/4/6 (which are vmnics 6/8/10/12) and port 1 has device functions 1/3/5/7 (which are vmnics 7/9/11/13).



The following shows the vSphere Storage Adapters Configuration page with four enabled iSCSI Offload protocol partitions (Broadcom iSCSI Adapter) vmhba32 through vmhba35. In this example, which is similar to the setup used for the user examples later in this document, Port 0's first partition (device function 0) is vmhba 32, Port 0's fourth partition (device function 6) is vmhba 34, Port 1's first partition (device function 1) is vmhba 33, and Port 1's fourth partition (device function 7) is vmhba 35.

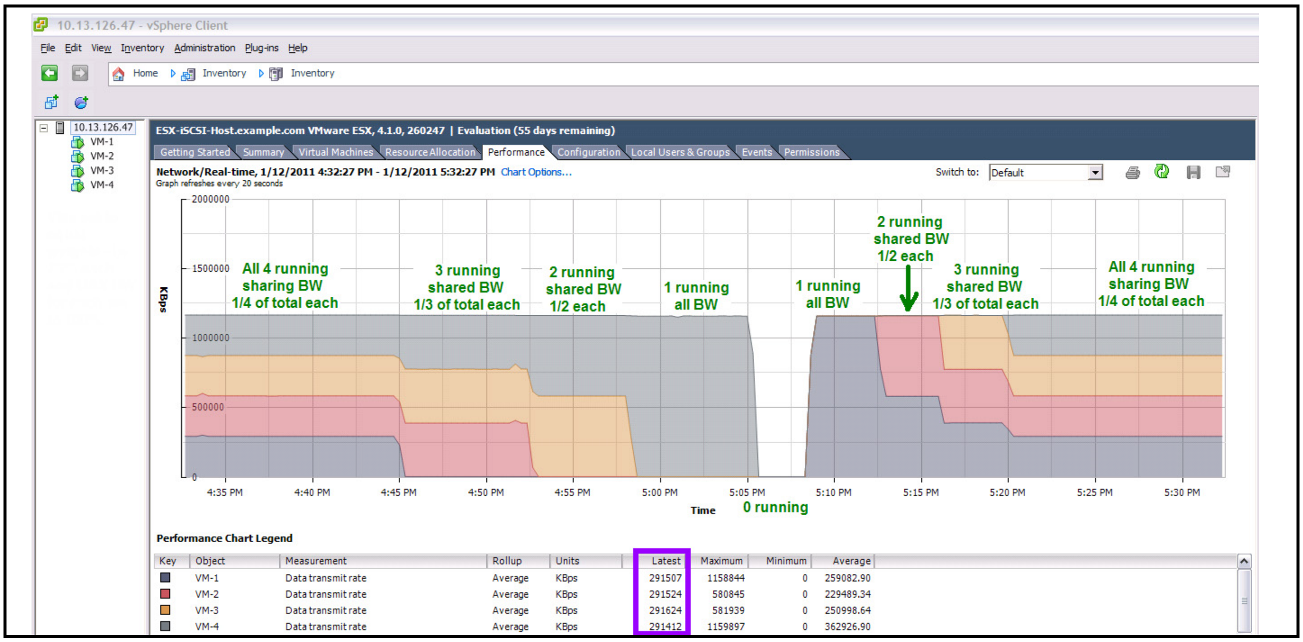

localhost. VMware ESXi, 4.1.0, 260247
Getting Started
Summary
Virtual Machines
Resource Allocation
Performance
Configuration
Local Users & Groups
Events
Permissions
Hardware
Health Status
Processors
Memory
Storage
Networking
Storage Adapters
Network Adapters
Advanced Settings
Power Management
Software
Storage Adapters
Device
Type
WWN
Broadcom iSCSI Adapter
vmhba32 iSCSI iqn.1998-01.com.vmware:localhost:974021991:34:bnx2i-00101888e7a8
vmhba33 iSCSI iqn.1998-01.com.vmware:localhost.:902103589:33:bnx2i-00101888e7aa
vmhba34 iSCSI iqn.1998-01.com.vmware:localhost:974021991:37:bnx2i-00101888e7b4
vmhba35 iSCSI iqn.1998-01.com.vmware:localhost.:902103589:35:bnx2i-00101888e7b6
Details
vmhba32
Model: Broadcom iSCSI Adapter
iSCSI Name: iqn.1998-01.com.vmware:localhost:974021991:34
iSCSI Alias: bnx2i-00101888e7a8

These Ethernet Protocol adapters in each partition are configurable like a normal port adapter in vSphere's Networking Configuration pages as shown below.

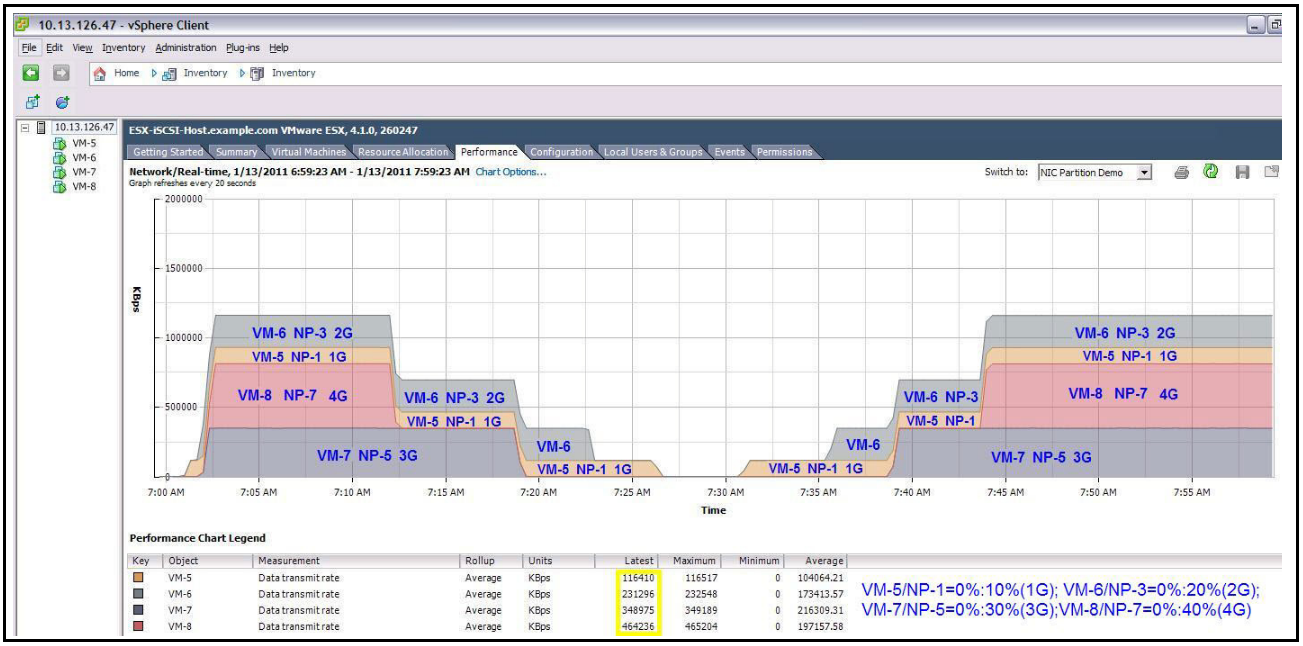


VMWare's vSphere 4.1 (or vCenter) can be used to view a selected VM's Networking performance in the specific Host's Performance subtab selecting. In the first example, the first port's four partitions (on VMNIC6/8/10/12) are set to 0% Relative Bandwidth Weight and 100% Maximum Bandwidth each and the second port's partitions (on VMNIC7/9/11/13) are similarly set to 0% Relative Bandwidth Weight but the Maximum Bandwidth values are set to 10%/20%/30%/40% respectively which results in VMNIC7's link speed (for the transmit direction only) indicating 1000 Mbps, VMNIC9's link speed indicating 2000 Mbps, VMNIC11's link speed indicating 3000 Mbps and finally VMNIC13's link speed indicating 4000 Mbps. This is indicated in the vSphere Host's Configuration - Network Adapter page.

The first port's network performance indicated in the vSphere Host's Performance page shows each individual VM's send traffic rate, when none to all four are sending, sharing the available bandwidth between each other.



The second port's network performance indicates each VM is limited to it's specific top end setting (Maximum Bandwidth) and does not expand into the unused area.

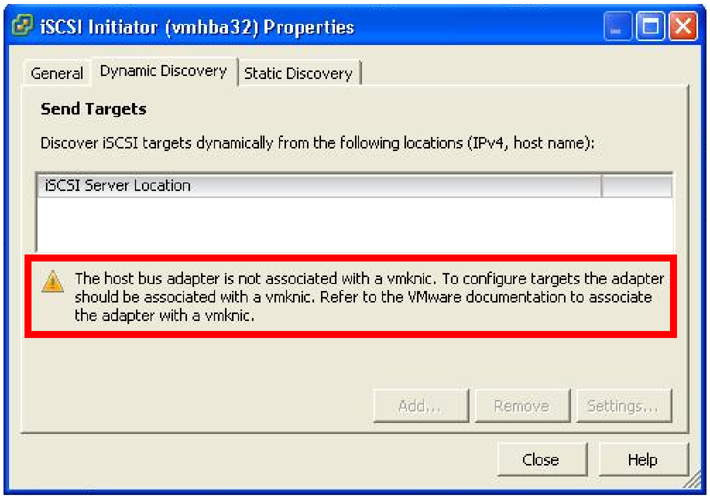


VMWare driver versions can be checked using the following commands:

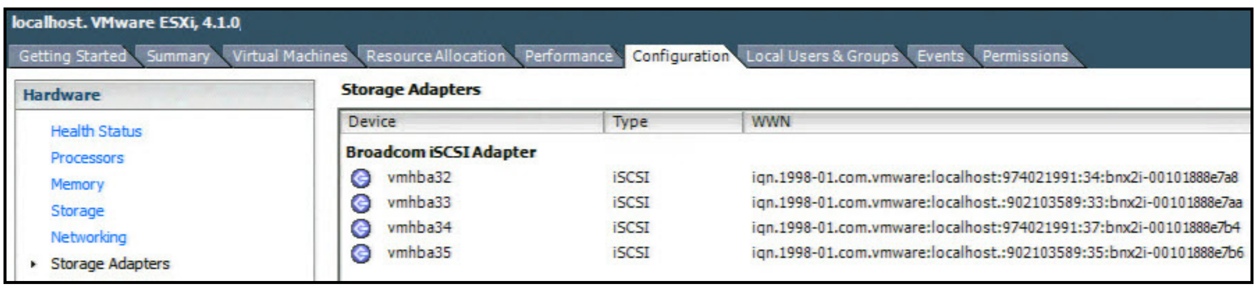
```
# esxcli software vib list
# vmkload_mod -s bnx2x
# vmkload_mod -s cnic
# vmkload_mod -s bnx2i
# vmkload_mod -s bnx2fc
# vm-support
# ethtool -i vmnicN
```

Details on Configuring iSCSI Offload on ESX/ESXi 4.1

After creating a networking vSwitch, but before attempting to discover any iSCSI targets, you need to associate any Broadcom (5709/5771x/578xx) iSCSI hardware offload HBA (vmhba) to a previously created vSwitch (vmk/vmnic). Otherwise, you will see a caution message when you attempt to discover any iSCSI targets via the Broadcom iSCSI Offloaded pathway. This association step is not required if using the non-offload software iSCSI pathway.



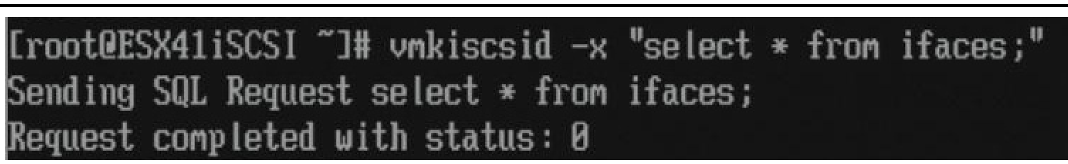
The Broadcom iSCSI Adapter's vmhba# are listed in the vSphere Configuration tab Hardware Storage Adapters window.



To associate or connect these iSCSI offload vmhba#'s to an existing vSwitch (vmk#/vmnic#), use the following ESX commands on the Host to list the current associated iSCSI initiators:

```
vmkiscsid -x "select * from ifaces;" | more
```

This lists all currently active and previously assigned inactive associations. If there are a lot of associations, use the "| more" to make the output easier to read. Since this is before associating any adapters, there is nothing listed in this example.



Use the following command to list which vmnic# can be associated with a specific vmhba#—which in this example are the numbers 32 through 35.

```
esxcli swiscsi vmnic list -d vmhba32
```

```
~ # esxcli swiscsi vmnic list -d vmhba32
vmnic6
    vmnic name: vmnic6
    mac address: 00:10:18:88:e7:a8
    mac address settable: NO
    maximum transfer rate: 10000
    current transfer rate: 10000
    maximum frame size: 1500
```

This command is a good way to determine which vmnic#'s (and their associated vSwitch vmk#) are assignable to each iSCSI Offload vmhba#'s.

The following command lists the information on previously associated vmhba# devices:
`esxcli swiscsi nic list -d vmhba`**`32`**

```
~ # esxcli swiscsi nic list -d vmhba32
vmk1
    pNic name: vmnic6
    ipv4 address: 1.1.41.1
    ipv4 net mask: 255.255.0.0
    ipv6 addresses:
    mac address: 00:10:18:88:e7:a8
    mtu: 1500
    toe: false
    tso: true
    tcp checksum: false
    vlan: true
    vlanId: 0
    ports reserved: 63488~65536
    link connected: true
    ethernet speed: 10000
    packets received: 13295
    packets sent: 15
    NIC driver: bnx2x
    driver version: 1.64.32.v41.1
    firmware version: 0 bc 6.4.18 phy aa0.511
```

This returns an error if the vmhba# does not have a vSwitch vmnic#/vmk# currently assigned to it.

```
~ # esxcli swiscsi nic list -d vmhba33
Errors:
No nics found for this adapter.
```

The following associates the desired iSCSI offload adapter vmhba# to the desired vSwitch vmk#/vmnic#. The vSwitches were all created earlier in the networking window in vSphere.
`esxcli swiscsi nic add --adapter vmhba# --nic vmk#`

or
`esxcli swiscsi nic add -d vmhba# –n vmk#`

```
~ # esxcli swiscsi nic add -d vmhba32 -n vmk1
~ # esxcli swiscsi nic add -d vmhba33 -n vmk2
~ # esxcli swiscsi nic add -d vmhba34 -n vmk7
~ # esxcli swiscsi nic add -d vmhba35 -n vmk8
```

If the wrong information is entered, an error message is displayed:

```
[root@ESX-iSCSI-Host ~]# esxcli swiscsi nic add -d vmhba34 -n vmk1
Errors:
Failed to add nic.
```

In this example, the Broadcom iSCSI adapter (vmhba32) on vmnic2 is associated with the vSwitch vmk0. Rerunning the previous ifaces list command shows the added iSCSI Offload device as associated with vmk0/ vmnic2.

```
[root@ESX41iSCSI ~]# esxcli swiscsi nic add --adapter vmhba32 --nic vmk0
[root@ESX41iSCSI ~]# vmkiscsid -x "select * from ifaces;"
Sending SQL Request select * from ifaces;
Record 0:
   `key`='bnx2i-00237de88258@vmk0'
   `iface.iscsi_ifacename`='bnx2i-00237de88258@vmk0'
   `iface.net_ifacename`='vmk0'
   `iface.ipaddress`='(null)'
   `iface.hwaddress`='default'
   `iface.transport_name`='bnx2i-00237de88258'
   `iface.phydev`='vmnic2'
   `iface.vlan`='0'
   `iface.portFirst`='63488'
   `iface.portCount`='2048'
Request completed with status: 0
```

The following command disassociates (or removes) all iSCSI offload adapters from the ifaces list.

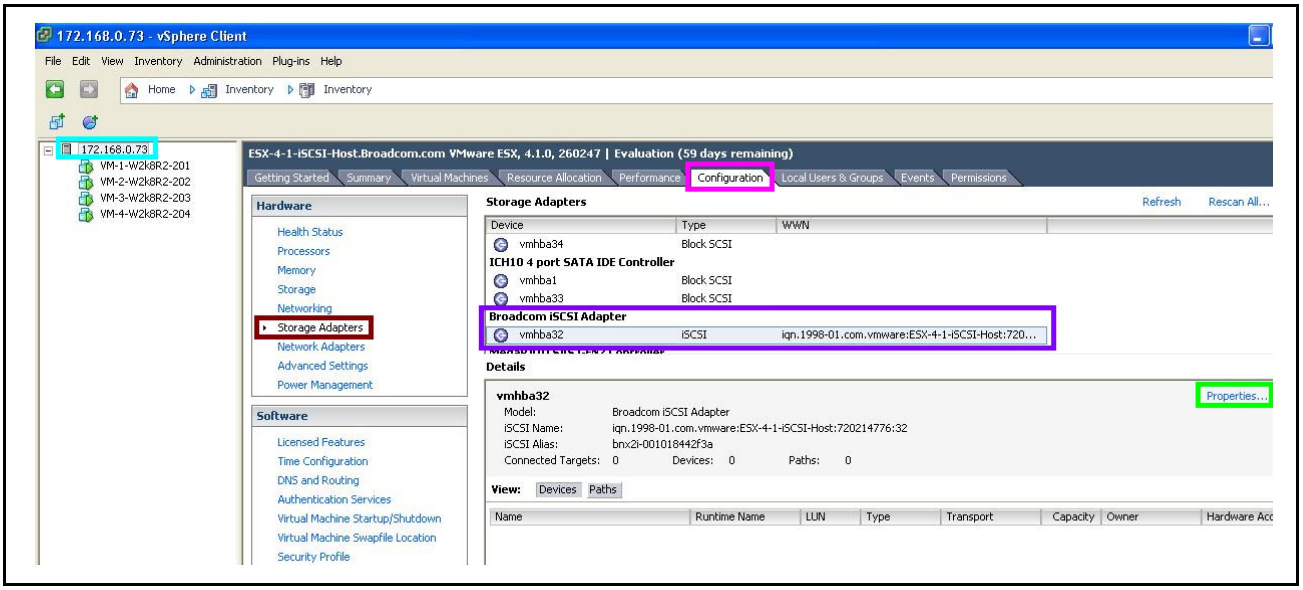
```
vmkiscsid -x "delete from ifaces;"
```

The following command can be used to delete a specific vmhba# association:

```
esxcli swiscsi nic remove -n <port_name> -d vmhba#
```

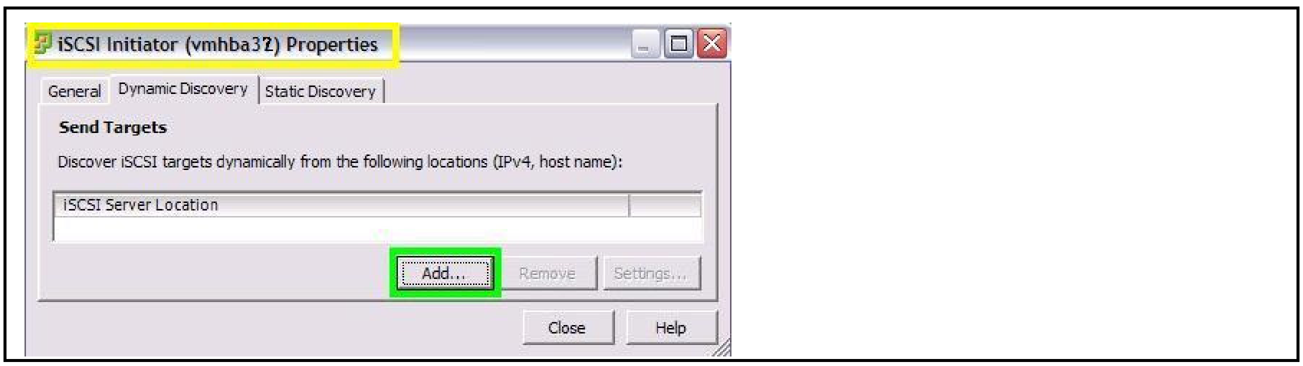
This command is used if changing the iSCSI HBA devices on your host.

Return to the vSphere Storage Adapters window and proceed with Discovering the iSCSI target LUNs. This is done by selecting the desired vmhba# in the Configuration Storage Adapters window and then clicking Properties in the Details window below it.

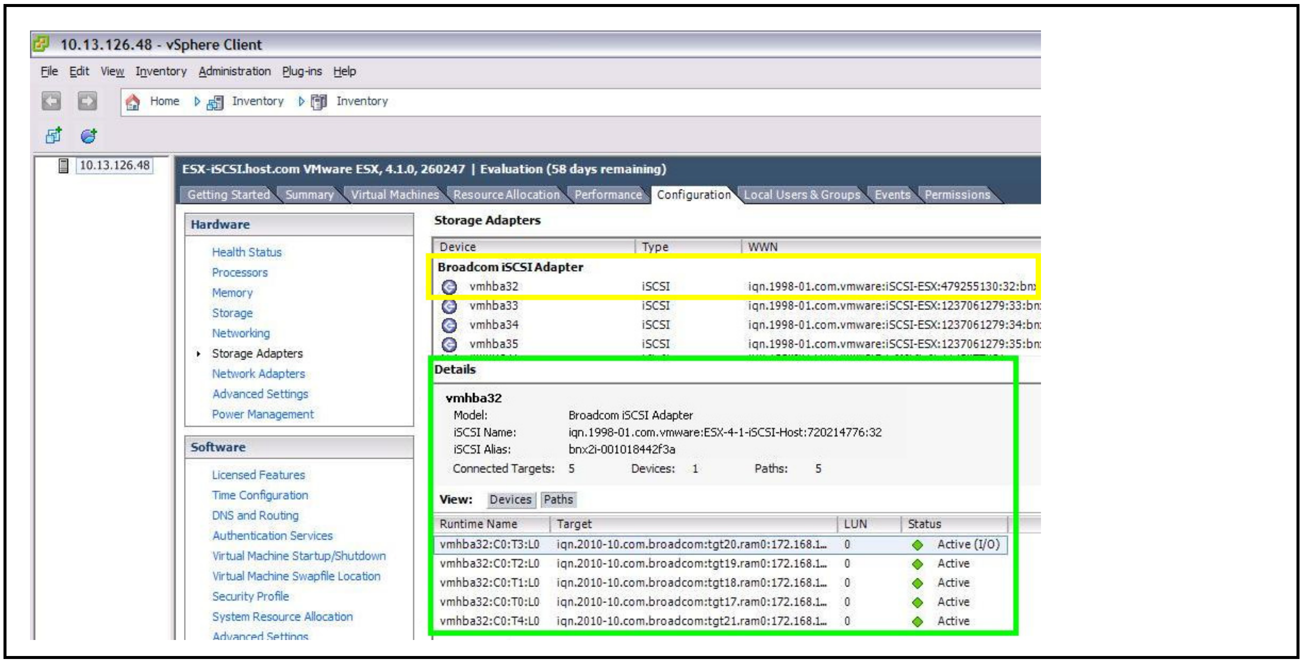


If the non-offload iSCSI pathway is needed, it must be enabled before use. This is done by clicking it's Properties in the Details window below it and enabling it by clicking the Configure button in the General tab.

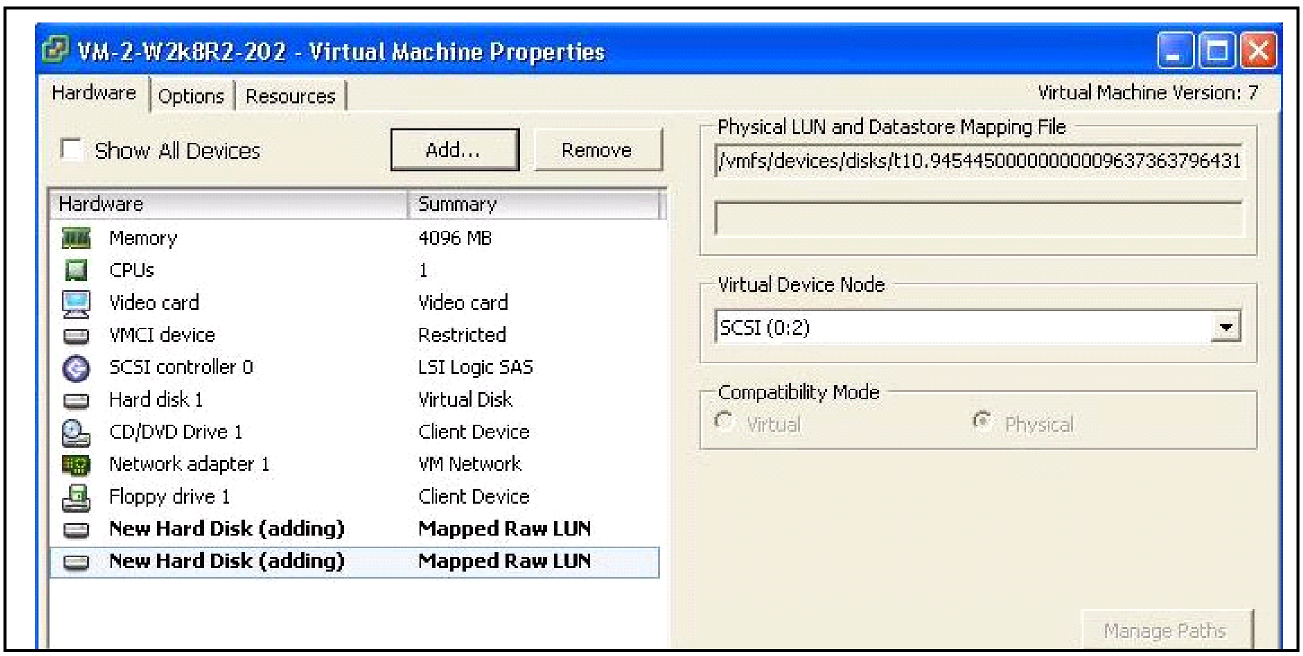
In the iSCSI Initiator (vmhba#) Properties window, select the Dynamic Discovery or Static Discovery tab and enter the desired iSCSI target LUN information.



After completing these Discovery steps, you are typically asked to refresh (rescan) the host bus adapter.



Now you can add the discovered LUNs as Hard Disks to the desired VMs.

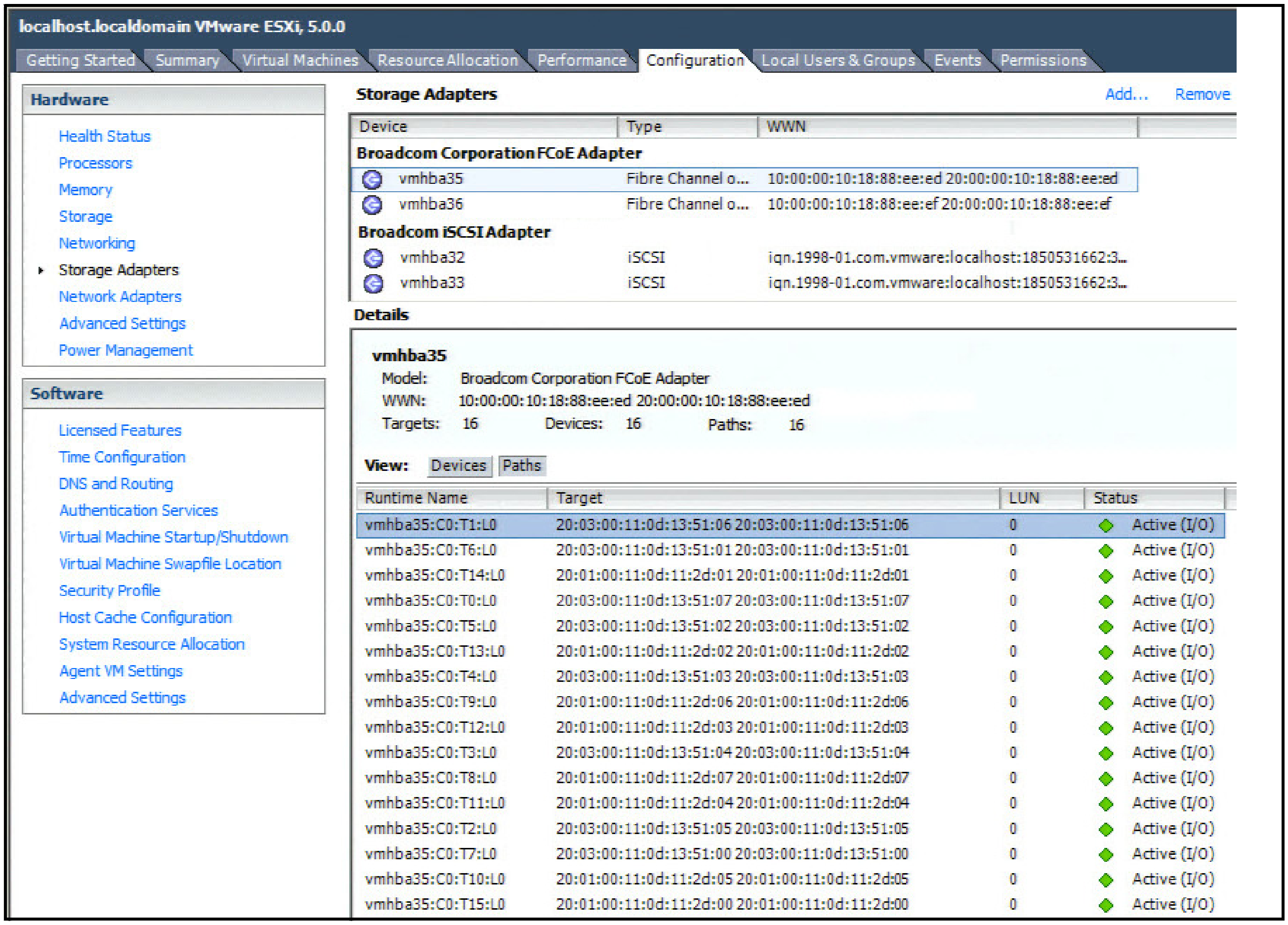


VMWare ESXi 5.0

VMWare ESXi 5.0 shows the respective device protocols that were enabled in USC if the respective NetXtreme II device drivers are installed. Go to the Dell Driver Download web site (http://support/del.com) for the latest device Firmware, if not already installed. Go to the VMWare web site (http://downloads.vmware.com) for the latest device drivers and ensure they are installed on your system. In VMWare, the Ethernet Protocol should be enabled on all eight partitions.

VMWare ESXi 5.0 supports the 57712-k DCB, L2 Ethernet, iSCSI Offload and FCoE Offload Protocols in both Switch Independent Partitioning and SF modes—in the VMWare driver versions 1.70.34.v50.1 (bnx2x), 2.70.1k.v50.2 (bnx2i), 1.0.1v.v50.1 (bnx2fc) and Dell firmware version 7.0.47 or later (see http://partnerweb.vmware.com/programs/hcl/ESX_NPAR_config.pdf). These same VMWare driver versions support the 578xx devices L2 Ethernet and iSCSI Offload Protocols in both Switch Independent Partitioning and SF modes. The 578xx devices support DCB, L2 Ethernet, iSCSI Offload and FCoE Offload Protocols in both Switch Independent Partitioning and SF modes with the VMWare driver versions 1.72.18.v50.4 (bnx2x), 2.72.10.v50.2 (bnx2i), 1.72.11.v50.1 (bnx2fc) and Dell firmware version 7.**2**.x or later.

The vSphere display on ESXi 5.0 is similar to ESX/ESXi 4.1, in that the Network Adapters window shows the same eight Broadcom Corporation NetXtreme II BCM57712 10 Gigabit Ethernet MultiFunction devices and the Storage Adapters window will show the same enabled iSCSI Offload devices. The newest feature of ESXi 5.0 is that the enabled FCoE Offload protocol devices are now visible. The following shows two FCoE Offload and two iSCSI Offload Adapters in the ESXi 5.0 vSphere Storage Adapters window.

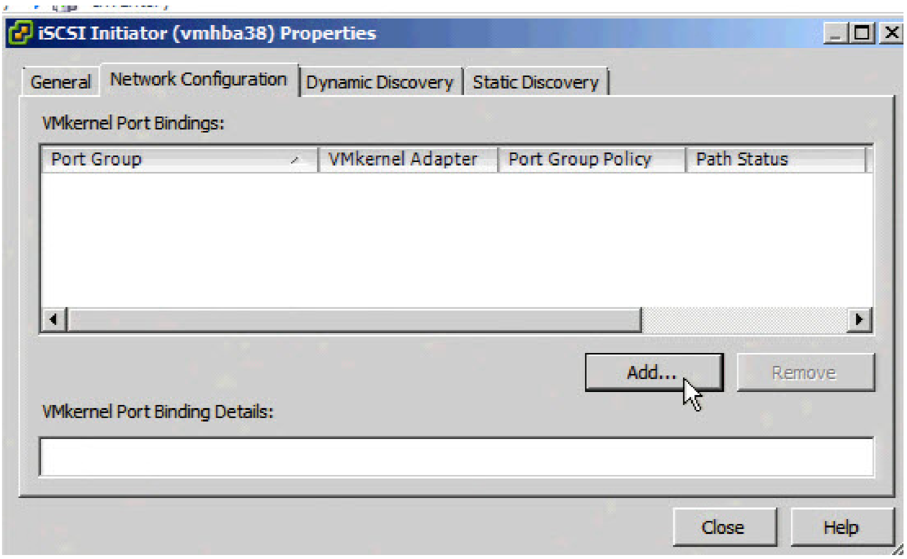


Note: VMWare ESX/ESXi 5.0 supports eight 10 GbE ports (see http://www.vmware.com/pdf/vsphere5/r50/vsphere-50-configuration-maximums.pdf). Using Switch Independent Partitioning mode allows you to expand the number of ports usable from eight physical ports to 32 virtual Switch Independent Partitioning ports simultaneously with up to two iSCSI Offload adapters per physical port (for up to 16 virtual Switch Independent Partitioning iSCSI devices) or one iSCSI Offload and one FCoE Offload adapter per physical port (for up to 8 virtual Switch Independent Partitioning iSCSI and eight virtual Switch Independent Partitioning FCoE devices). This allows better port flexibility, traffic isolation, service quality, and bandwidth tuning for your management/backup/migration/ production networks.

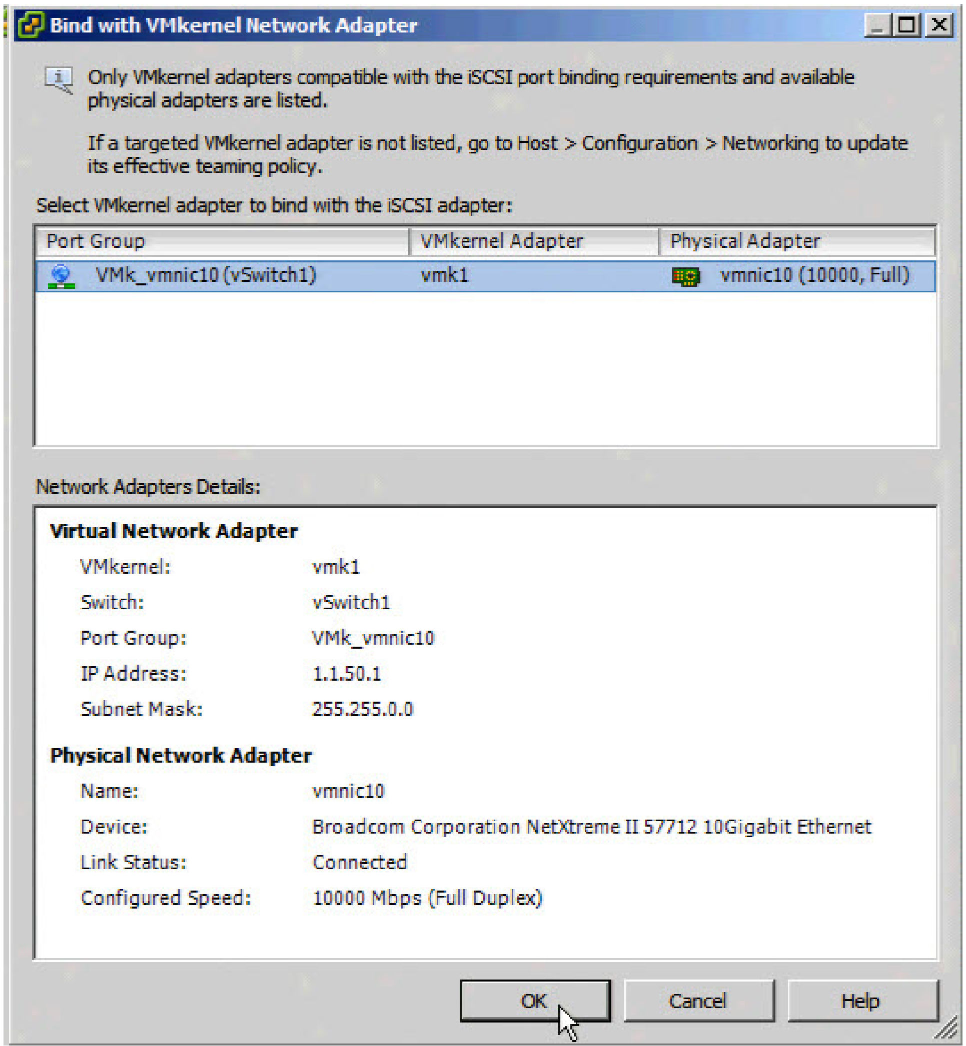
Details on Configuring iSCSI Offload on ESXi 5.0

After creating the desired networking vSwitches, you still need to associate the Broadcom (5709/5771x/578xx) iSCSI hardware offload HBA (vmhba#) devices to a previously created vSwitch (vmk#/vmnic#).

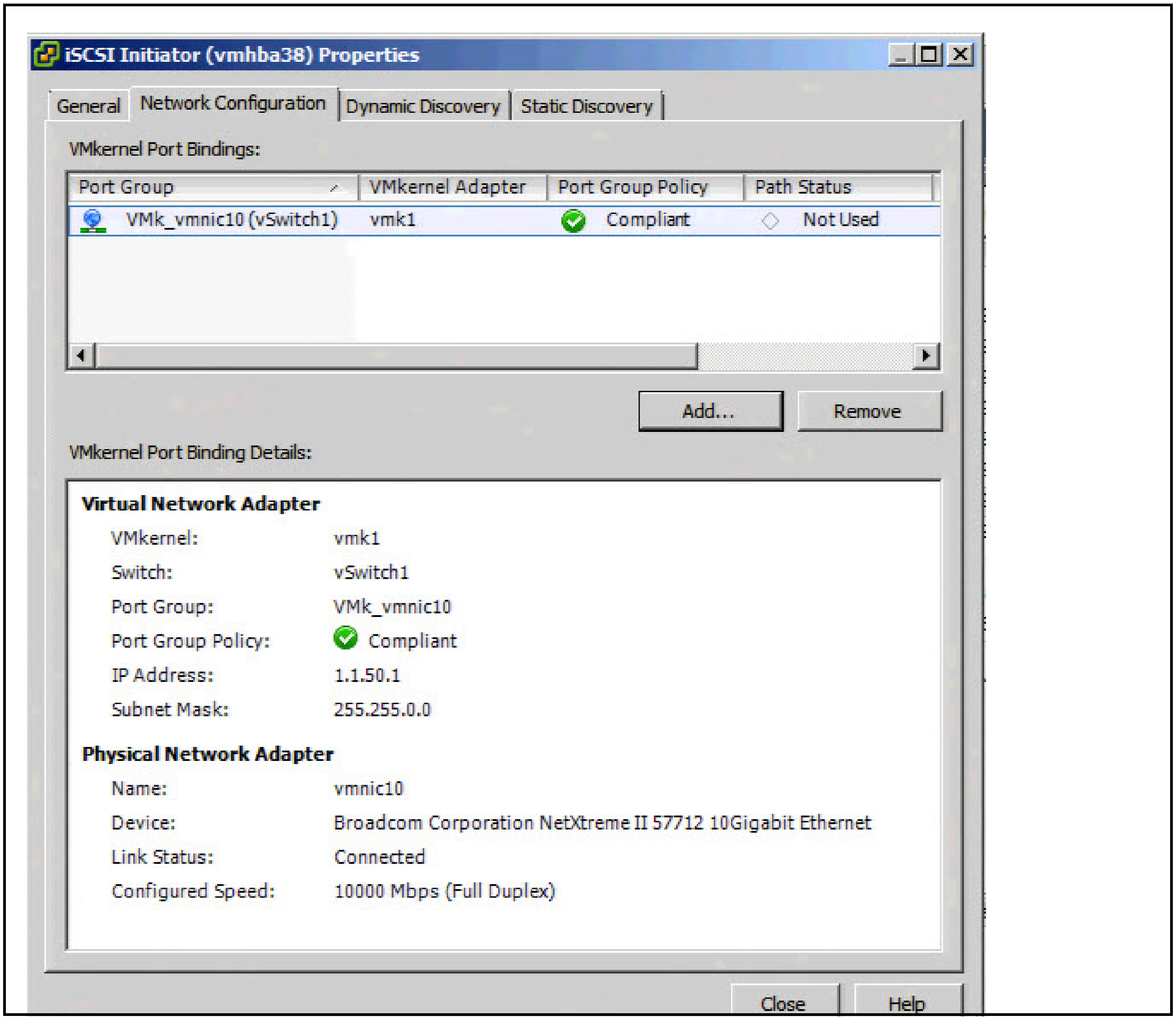
1. For the desired vmhba# in the Configuration Storage Adapters window, click **Properties** in the **Details** window below it.
2. Select the Networking Configuration tab and click the Add button.



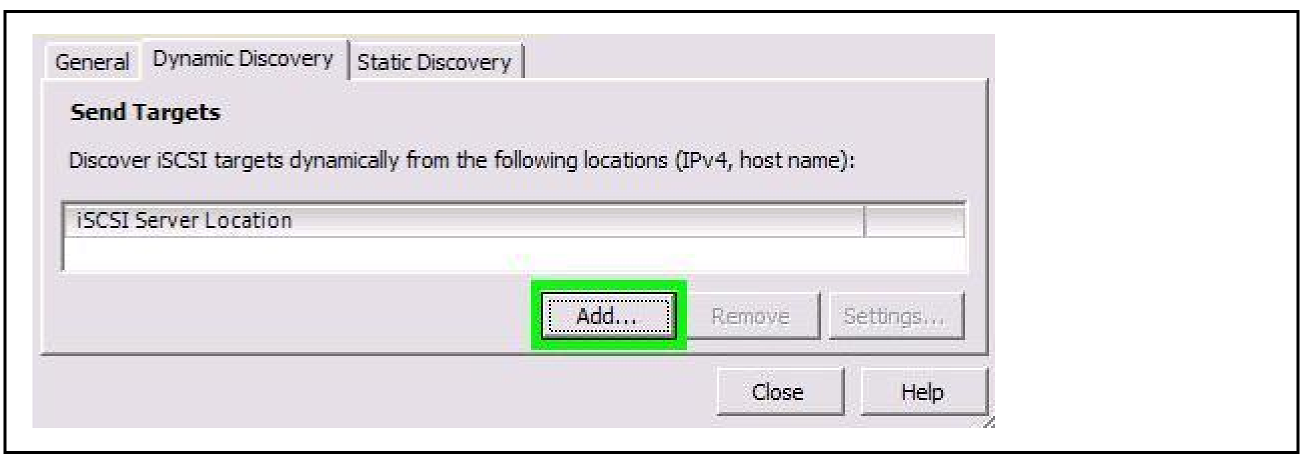
3. Select the VMKernel (vmk#) adapter that is compatible with this iSCSI Offload device for binding and click **OK**.



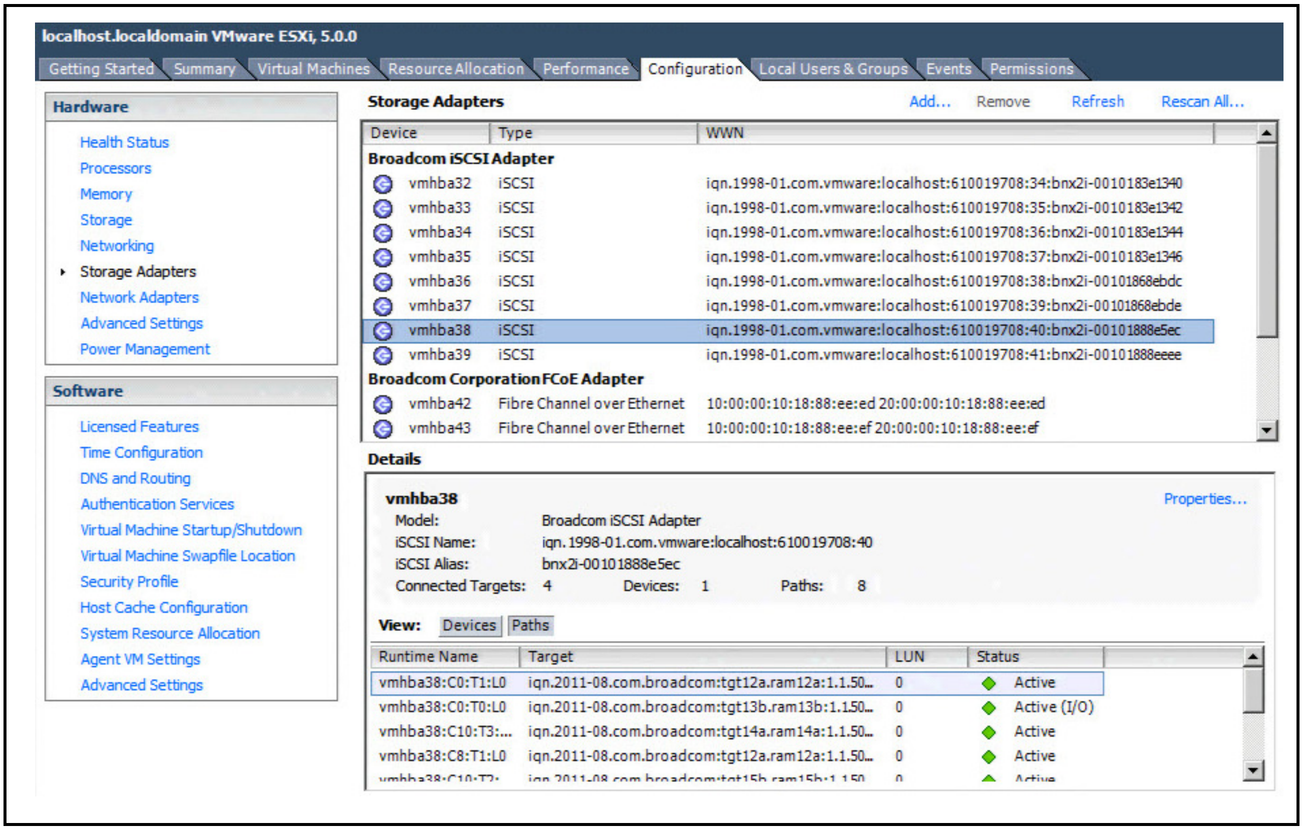
If successful, the vmk# displays that it is “Compliant”.



4. Go to the **Dynamic Discovery** or **Static Discovery** tab and enter the desired iSCSI target LUN information.



After completing these Discovery steps, you are typically asked to refresh (rescan) the host bus adapter. You can now add the discovered iSCSI LUNs as Hard Disks to the desired VMs.



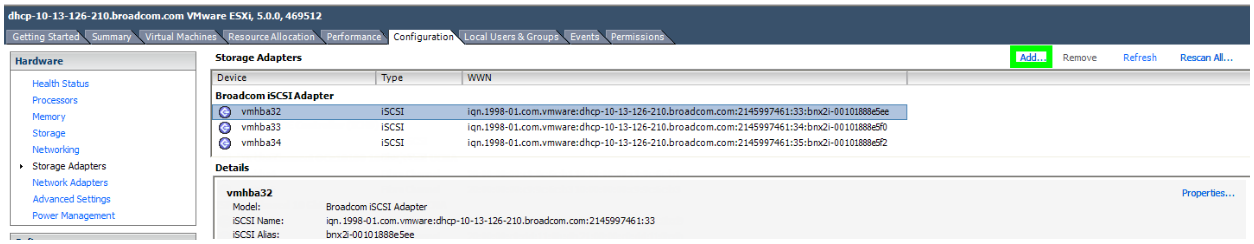
Note: VMWare's ESXi 5.x can experience boot up delays when there are remote iSCSI target connections present (over either offloaded hardware or non-offload software pathways). See http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2007108 for a description of the issue and workarounds.

Details on Configuring FCoE Offload on ESXi 5.0

After creating the desired networking vSwitches, you need to associate the Broadcom (57712/578xx) FCoE hardware offload HBA (vmhba#) devices to a previously created vSwitch (vmk#/vmnic#). This can be done two ways – via the command line (in SSH) or vSphere.

To use vSphere:

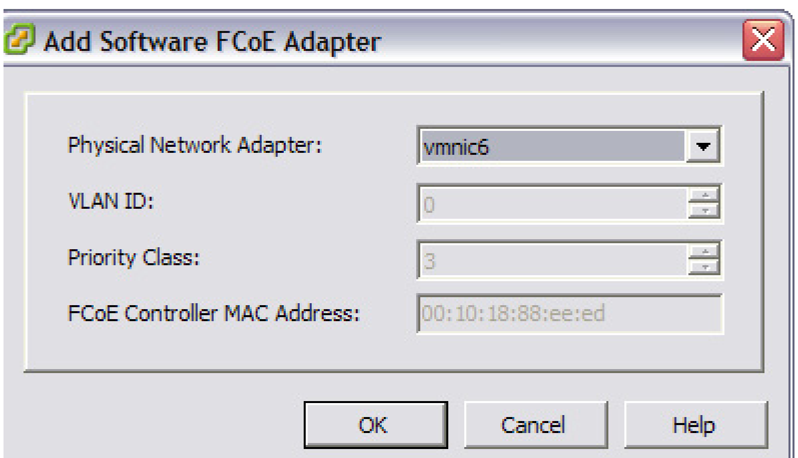
1. In ESXi 5.0 vSphere's Storage Adapters Configuration page, click **Add** in the upper right corner of the window.



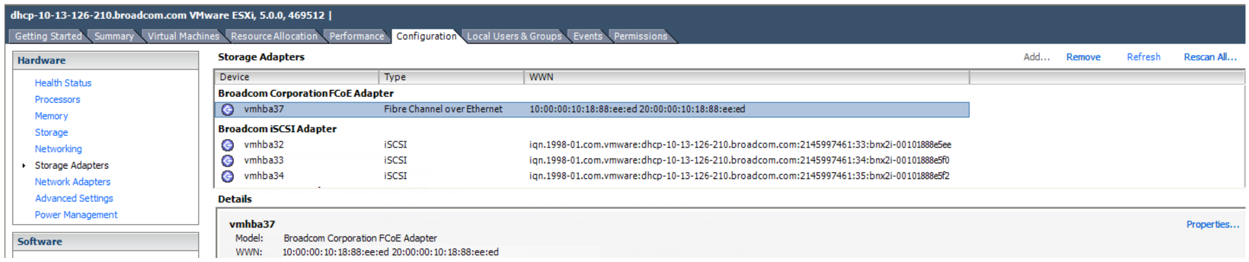
This is where you add either the iSCSI non-offload software initiator pathway or, in this case, the Broadcom FCoE Adapter. ESXi 5.0 lists all FCoE Adapters as Software even if Hardware Offloaded.

2. Select **Add Software FCoE Adapter** and click **OK**.

3. In the next window, select the desired 57712-k/578xx FCoE Hardware Offload adapter port and click **OK**.



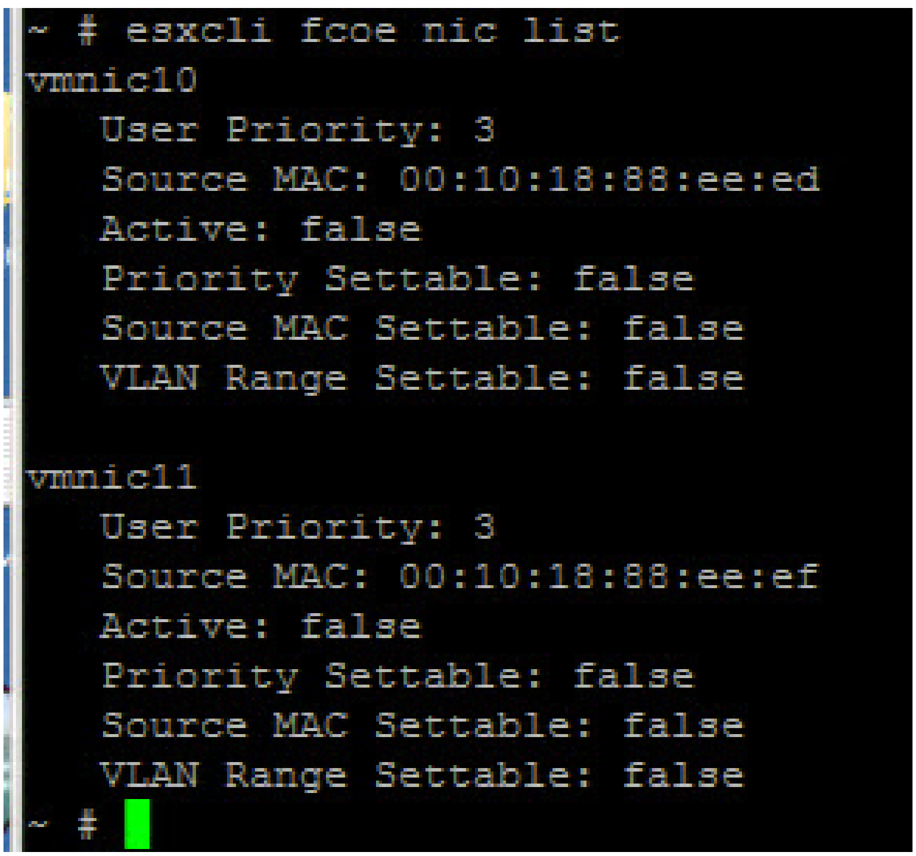
The new Broadcom FCoE Hardware Offloaded Adapter now appears in the Storage Adapter's list and is ready for use.



You can also use the command line command and an SSH connection to the ESXi 5.0 host to list the current FCoE offload ports — the ports may be inactive (Active: false) at first.

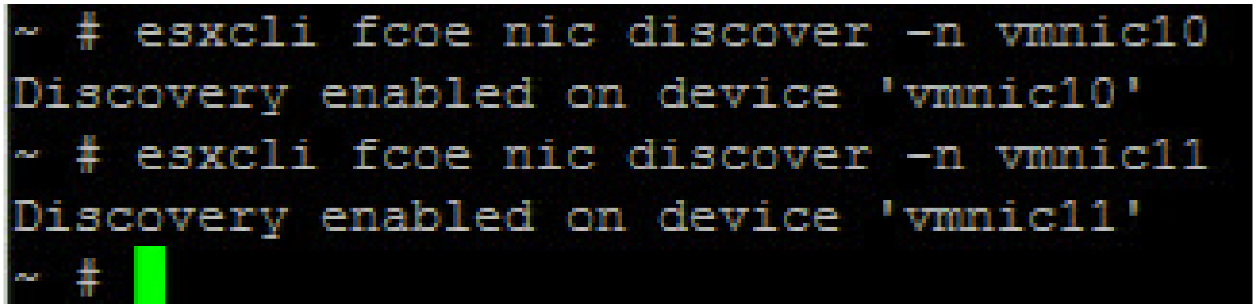
1. Enter the following command:

```
# esxcli fcoe nic list
vmnic#
   User Priority: 3
   Source MAC: xx:xx:xx:xx:xx:xx
   Active: false
   Priority Settable: false
   Source MAC Settable: false
   VLAN Range Settable: false
```



2. Use the following command to enable discovery (activate) on the selected FCoE ports:

```
…# esxcli fcoe nic discover -n vmnic#
```



```
~ # esxcli fcoe nic discover -n vmnic10
Discovery enabled on device 'vmnic10'
~ # esxcli fcoe nic discover -n vmnic11
Discovery enabled on device 'vmnic11'
~ #
```

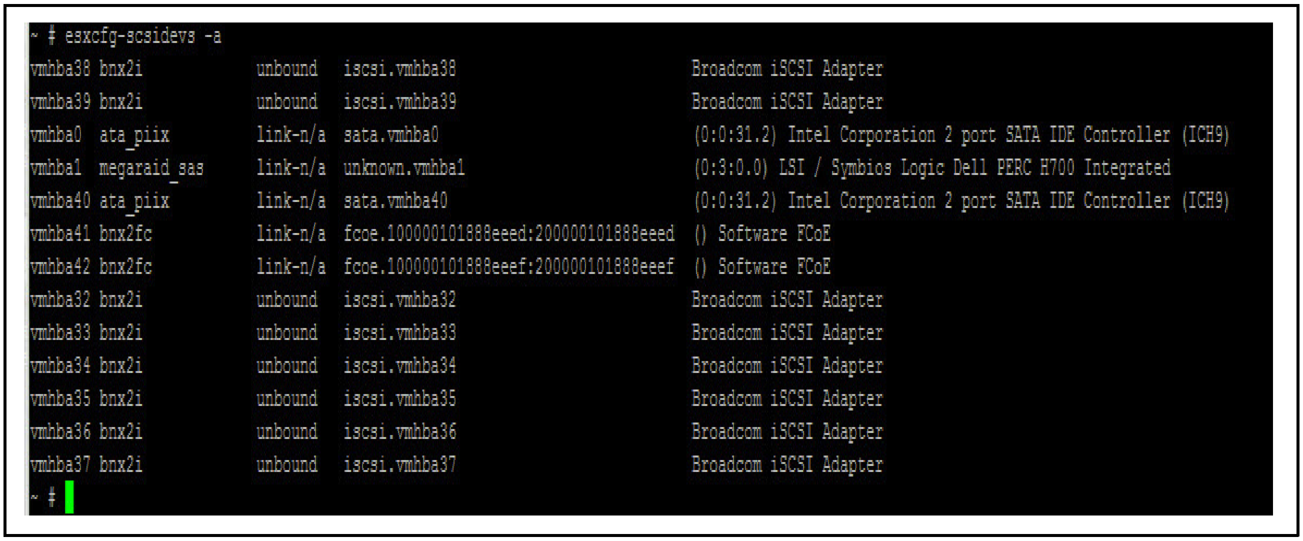
3. Rerun the previous FCoE list command (from Step 1) and you should see the ports now “active”.

```
# esxcli fcoe nic list
vmnic#
   User Priority: 3
   Source MAC: xx:xx:xx:xx:xx:xx
   Active: true
   Priority Settable: false
   Source MAC Settable: false
   VLAN Range Settable: false
```

```
~ # esxcli fcoe nic list
vmnic10
   User Priority: 3
   Source MAC: 00:10:18:88:ee:ed
   Active: true
   Priority Settable: false
   Source MAC Settable: false
   VLAN Range Settable: false

vmnic11
   User Priority: 3
   Source MAC: 00:10:18:88:ee:ef
   Active: true
   Priority Settable: false
   Source MAC Settable: false
   VLAN Range Settable: false
~ #
```

4. Use the following command to see the host's iSCSI and FCoE devices:

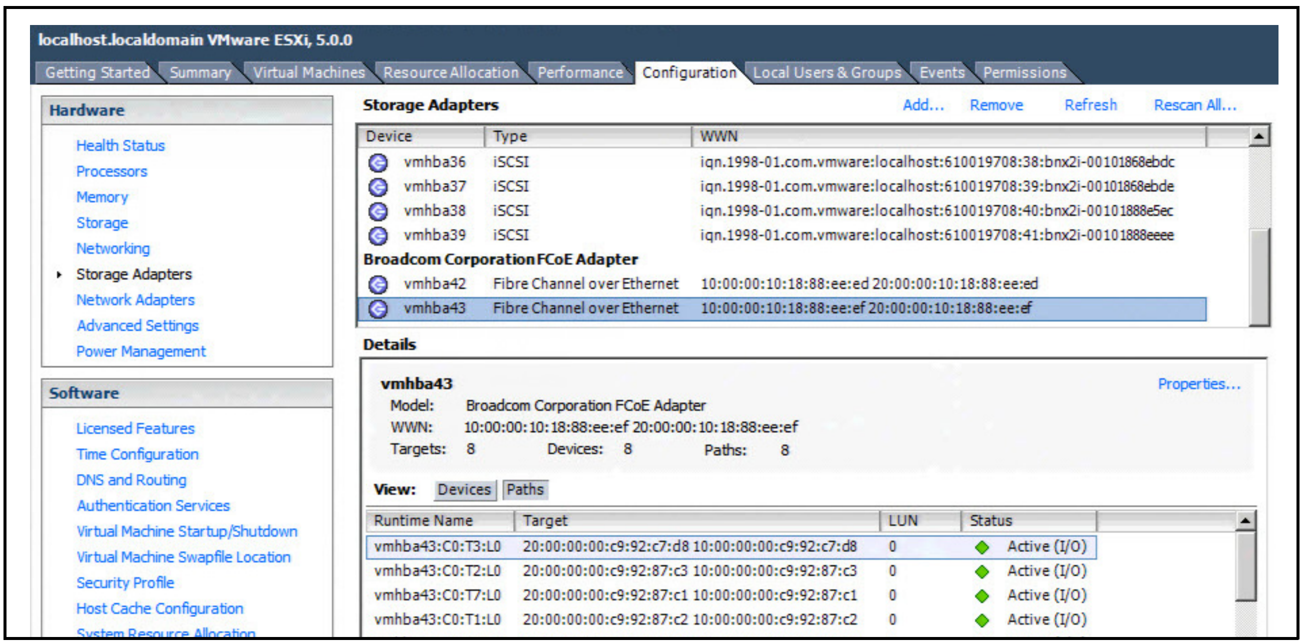
```
…# esxcfg-scsidevs -a
…
Vmhba# bnx2i  unbound  iscsi.vmhba#  Broadcom iSCSI Adapter
Vmhba# bnx2i  unbound  iscsi.vmhba#  Broadcom iSCSI Adapter
Vmhba# bnx2fc link-up  fcoe.xxx:xxx  () Software FCoE
Vmhba# bnx2fc link-up  fcoe.xxx:xxx  () Software FCoE
Vmhba# iscsi_vmk online iscsi.vmhba# iSCSI Software Adapter
```



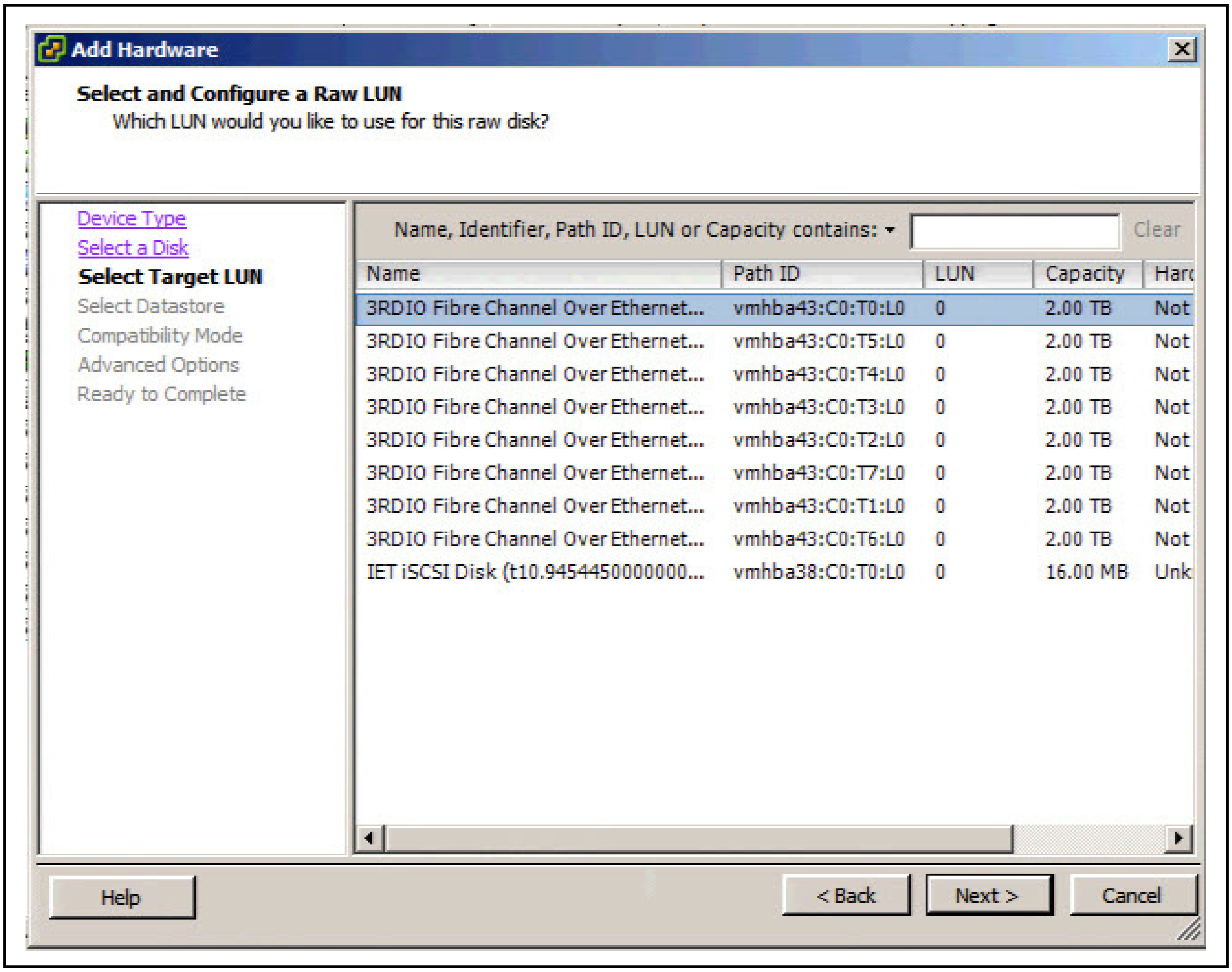
```
~ # esxcfg-scsidevs -a
vmhba38 bnx2i          unbound   iscsi.vmhba38                             Broadcom iSCSI Adapter
vmhba39 bnx2i          unbound   iscsi.vmhba39                             Broadcom iSCSI Adapter
vmhba0  ata_piix       link-n/a  sata.vmhba0                               (0:0:31.2) Intel Corporation 2 port SATA IDE Controller (ICH9)
vmhba1  megaraid_sas   link-n/a  unknown.vmhba1                            (0:3:0.0) LSI / Symbios Logic Dell PERC H700 Integrated
vmhba40 ata_piix       link-n/a  sata.vmhba40                              (0:0:31.2) Intel Corporation 2 port SATA IDE Controller (ICH9)
vmhba41 bnx2fc         link-n/a  fcoe.100000101888eeed:200000101888eeed   () Software FCoE
vmhba42 bnx2fc         link-n/a  fcoe.100000101888eeef:200000101888eeef    () Software FCoE
vmhba32 bnx2i          unbound   iscsi.vmhba32                             Broadcom iSCSI Adapter
vmhba33 bnx2i          unbound   iscsi.vmhba33                             Broadcom iSCSI Adapter
vmhba34 bnx2i          unbound   iscsi.vmhba34                             Broadcom iSCSI Adapter
vmhba35 bnx2i          unbound   iscsi.vmhba35                             Broadcom iSCSI Adapter
vmhba36 bnx2i          unbound   iscsi.vmhba36                             Broadcom iSCSI Adapter
vmhba37 bnx2i          unbound   iscsi.vmhba37                             Broadcom iSCSI Adapter
~ #
```

Note that the FCoE Offload device is listed as “Software FCoE” by the command’s output.

At this point, the FCoE LUNs will show up IF the connected FCF switch has been setup for this initiator to connect to the desired FC or FCoE target LUNs.

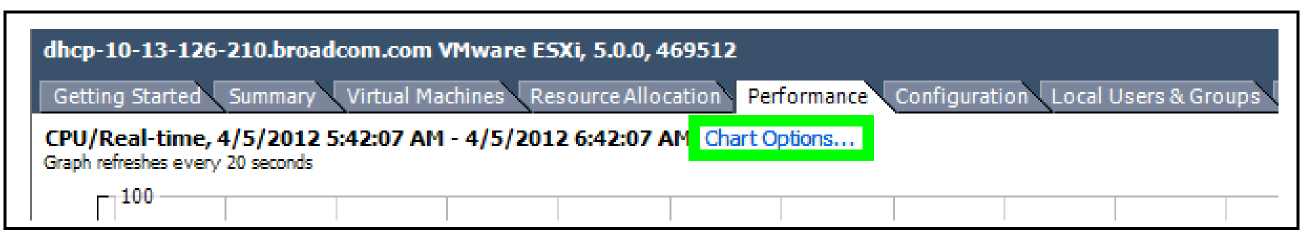


5. Add the discovered iSCSI and/or FCoE LUNs as Hard Disks to the desired VMs.

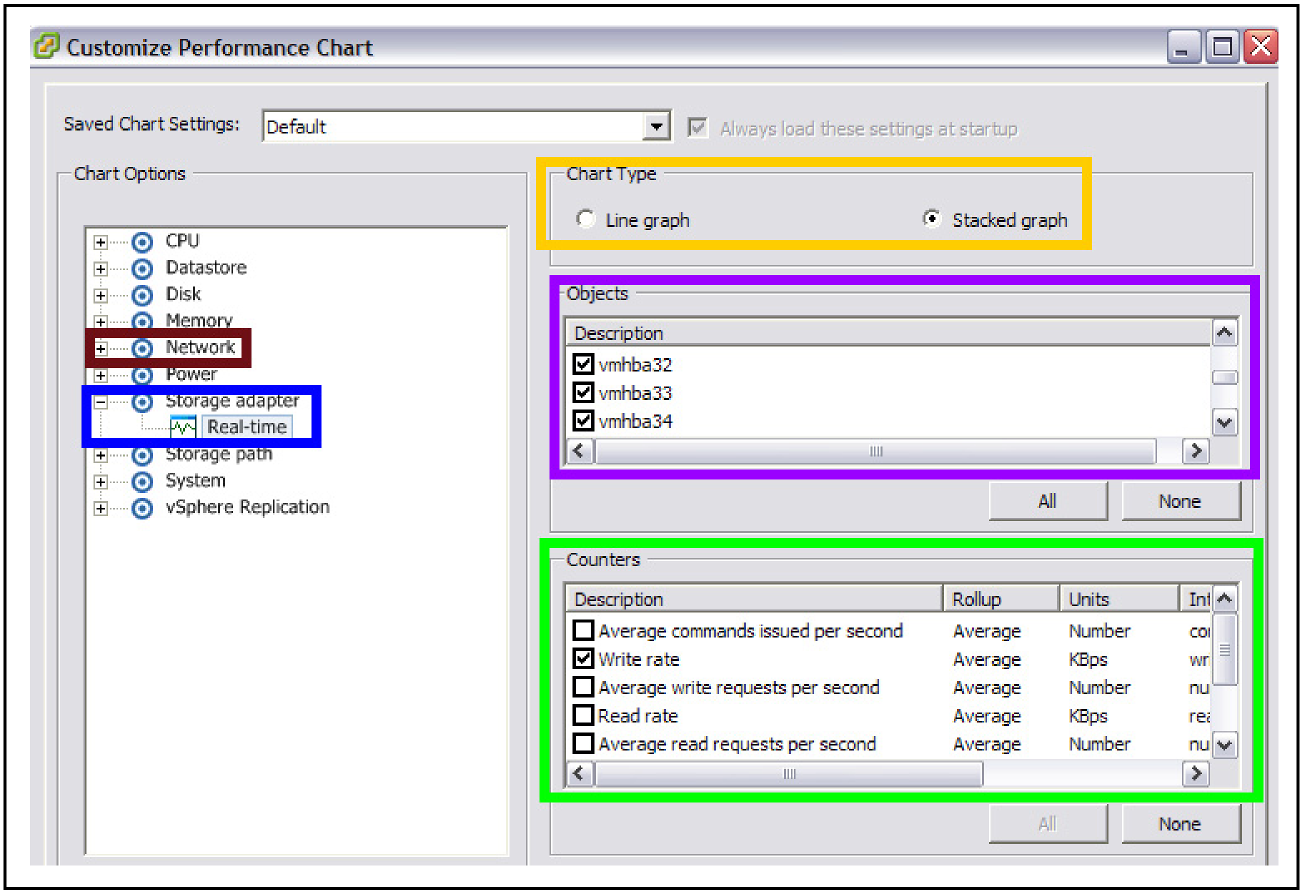


Details on Viewing Networking and Storage Performance on ESX

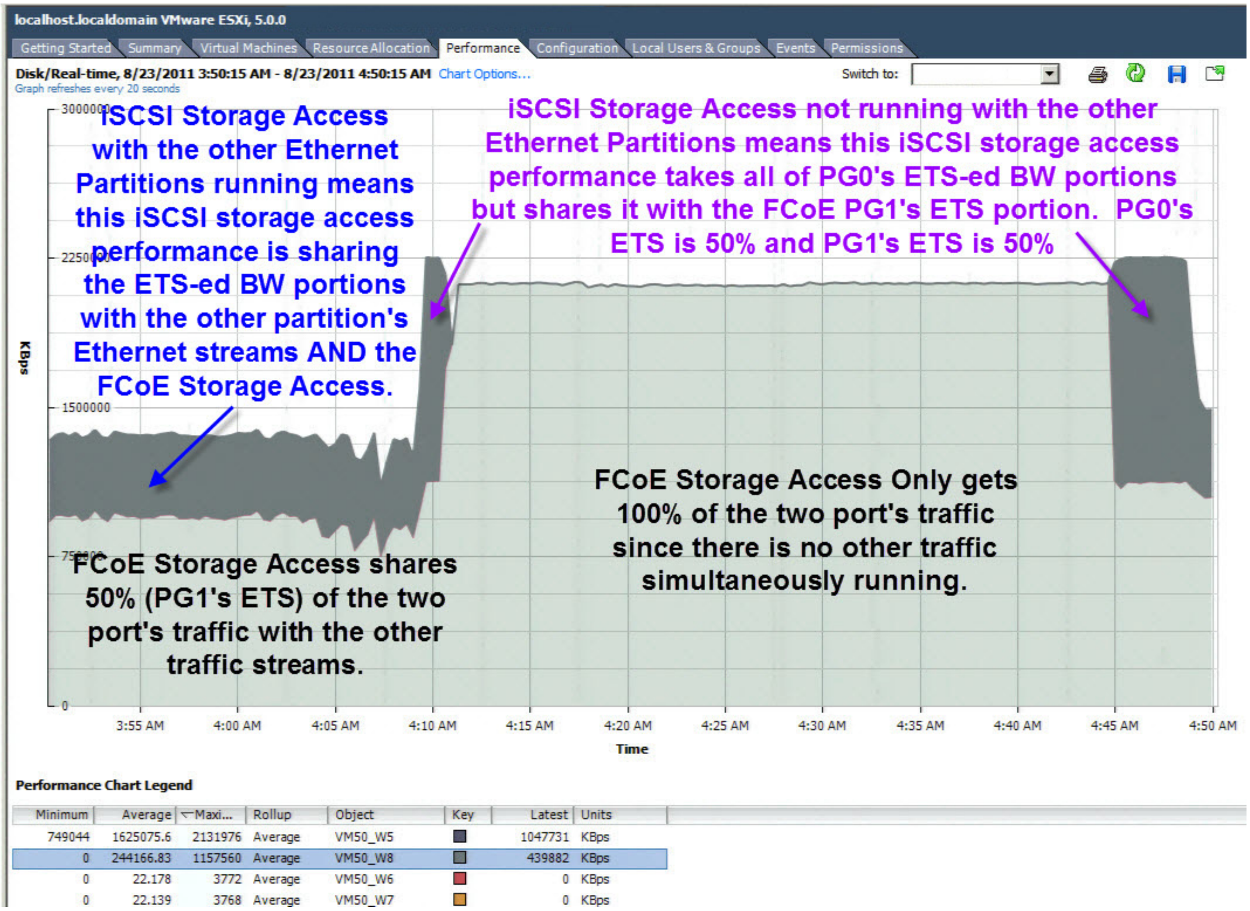
1. To view a port or partitions performance, go to ESX/ESXi 4.1 or ESXi 5.0 vSphere's Performance tab and open the Chart Options.

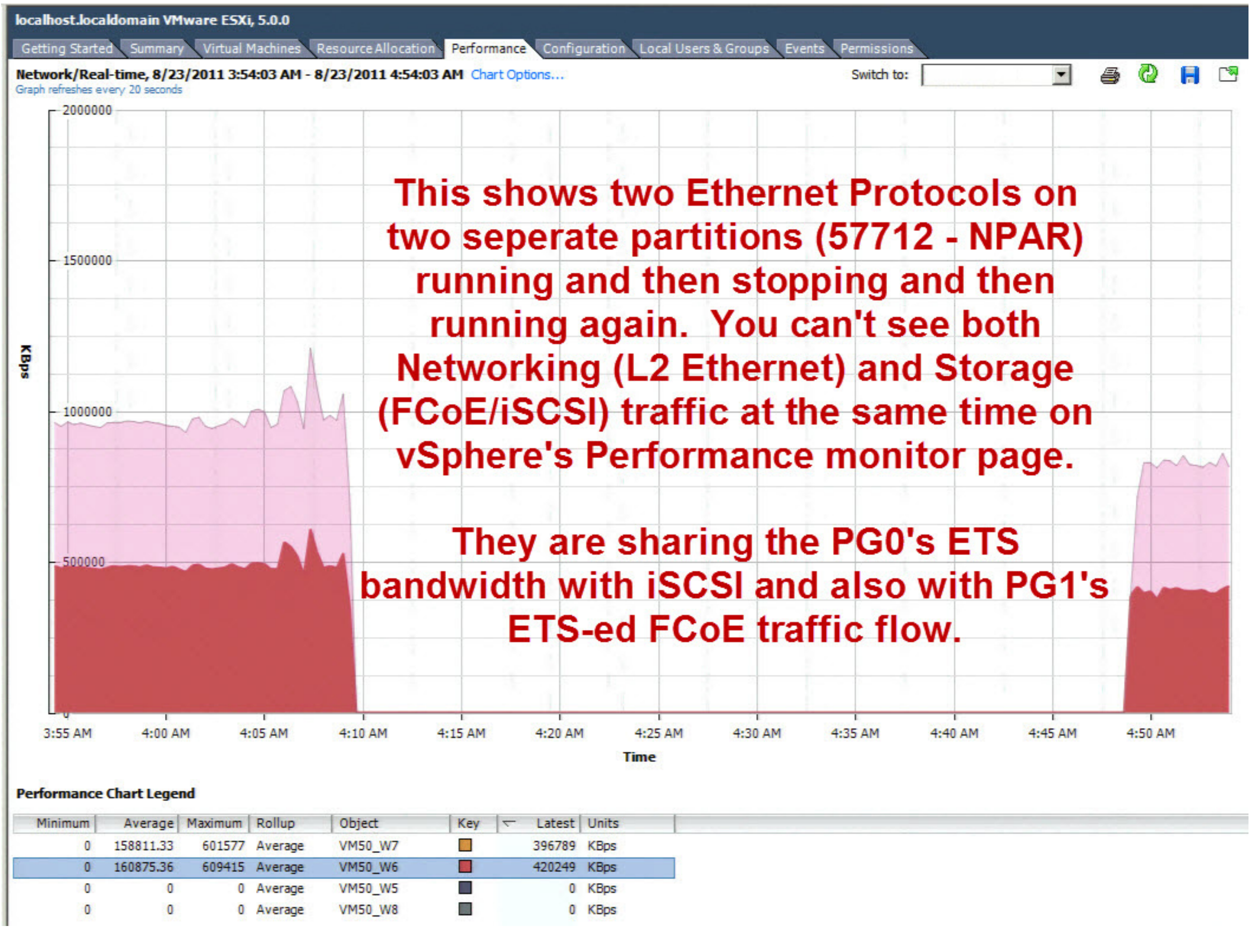


2. In the next window, select either Networking for L2 Ethernet OR Storage Adapter for iSCSI/FCoE and:
   - Select either line or graph.
   - Select which ports/partitions to view.
   - Select what counters to display.



The following screen captures show the 57712-k Converged NDC in DCB mode with mixed L2 Ethernet, iSCSI and FCoE traffic running at various points in both the ESX Performance Networking and Storage Adapter displays.


localhost.localdomain VMware ESXi, 5.0.0
Getting Started
Summary
Virtual Machines
Resource Allocation
Performance
Configuration
Local Users & Groups
Events
Permissions
Disk/Real-time, 8/23/2011 3:50:15 AM - 8/23/2011 4:50:15 AM
Chart Options...
Graph refreshes every 20 seconds
Switch to:
iSCSI Storage Access with the other Ethernet Partitions running means this iSCSI storage access performance is sharing the ETS-ed BW portions with the other partition's Ethernet streams AND the FCoE Storage Access.
iSCSI Storage Access not running with the other Ethernet Partitions means this iSCSI storage access performance takes all of PG0's ETS-ed BW portions but shares it with the FCoE PG1's ETS portion. PG0's ETS is 50% and PG1's ETS is 50%
FCoE Storage Access Only gets 100% of the two port's traffic since there is no other traffic simultaneously running.
FCoE Storage Access shares 50% (PG1's ETS) of the two port's traffic with the other traffic streams.
KBps
3000000
2250000
1500000
0
3:55 AM
4:00 AM
4:05 AM
4:10 AM
4:15 AM
4:20 AM
4:25 AM
4:30 AM
4:35 AM
4:40 AM
4:45 AM
4:50 AM
Time
Performance Chart Legend
Minimum
Average
Maxi...
Rollup
Object
Key
Latest
Units
749044 1625075.6 2131976 Average VM50_W5 1047731 KBps
0 244166.83 1157560 Average VM50_W8 439882 KBps
0 22.178 3772 Average VM50_W6 0 KBps
0 22.139 3768 Average VM50_W7 0 KBps


localhost.localdomain VMware ESXi, 5.0.0
Getting Started
Summary
Virtual Machines
Resource Allocation
Performance
Configuration
Local Users & Groups
Events
Permissions
Network/Real-time, 8/23/2011 3:54:03 AM - 8/23/2011 4:54:03 AM
Chart Options...
Graph refreshes every 20 seconds
Switch to:
2000000
1500000
1000000
500000
0
KBps
This shows two Ethernet Protocols on two seperate partitions (57712 - NPAR) running and then stopping and then running again. You can't see both Networking (L2 Ethernet) and Storage (FCoE/iSCSI) traffic at the same time on vSphere's Performance monitor page.
They are sharing the PG0's ETS bandwidth with iSCSI and also with PG1's ETS-ed FCoE traffic flow.
3:55 AM
4:00 AM
4:05 AM
4:10 AM
4:15 AM
4:20 AM
4:25 AM
4:30 AM
4:35 AM
4:40 AM
4:45 AM
4:50 AM
Time
Performance Chart Legend
Minimum
Average
Maximum
Rollup
Object
Key
Latest
Units
0 158811.33 601577 Average VM50_W7 396789 KBps
0 160875.36 609415 Average VM50_W6 420249 KBps
0 0 0 Average VM50_W5 0 KBps
0 0 0 Average VM50_W8 0 KBps

Setting MTU Sizes

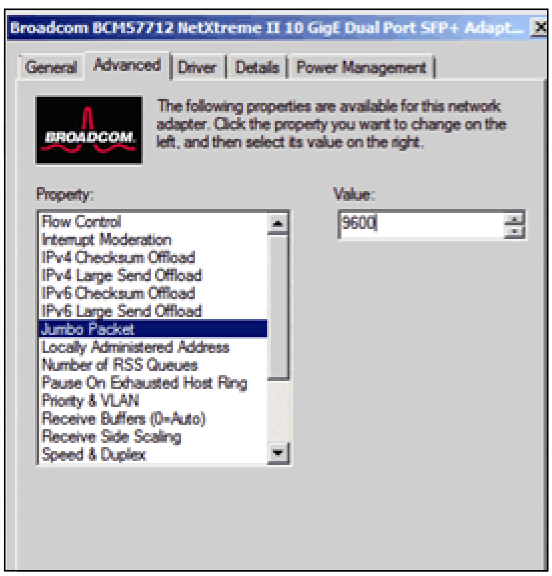
- Setting MTU Sizes in Windows
- Setting MTU Sizes in Linux
- Setting MTU Sizes in VMWare ESX/ESXi 4.1
- Setting MTU Sizes in VMWare ESXi 5.0

Note: In all cases, the connecting switch port that a 57712-k Switch Independent Partitioning port is connected to must have the switch's MTU size set to the largest MTU size of those four partitions of the port if the user wants to support all four partitions MTU size settings. Additionally, the remaining network that the traffic flows though must also support the desired MTU sizes for that sized frames to be used without being dropped/truncated/fragmented by the network.

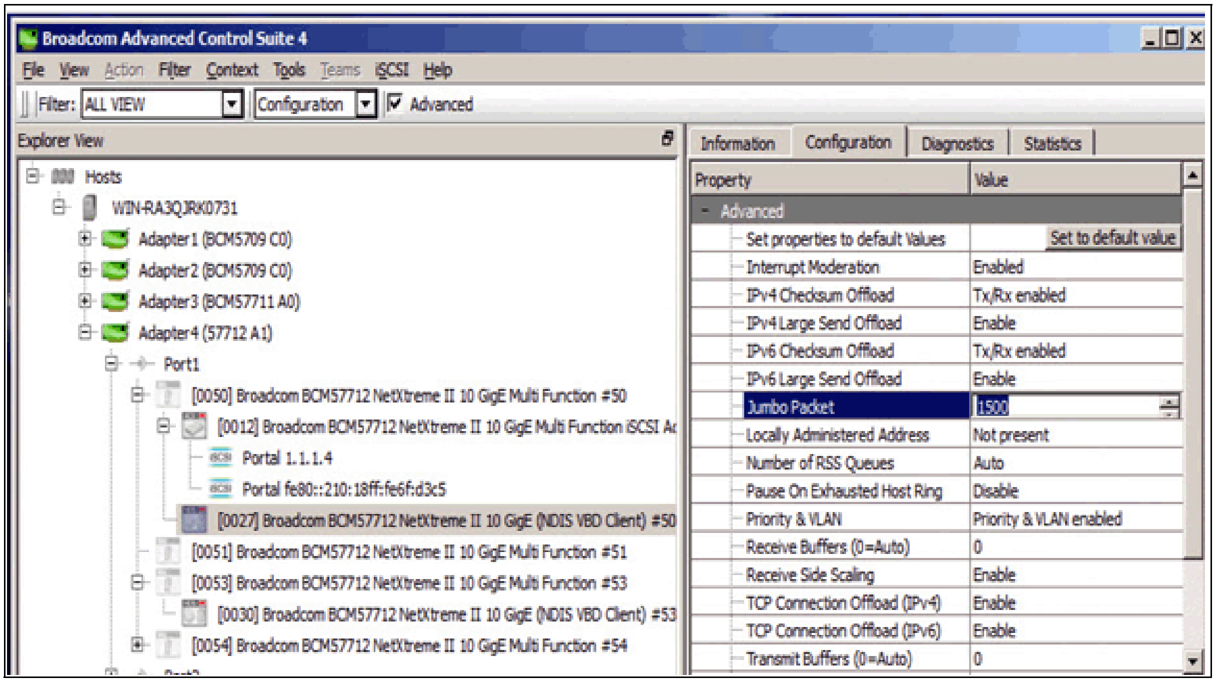
Setting MTU Sizes in Windows

The MTU size for each individual Ethernet protocol-enabled partition can be independently set from Normal (1500 bytes) up to Jumbo (9600 bytes) in several places in Windows.

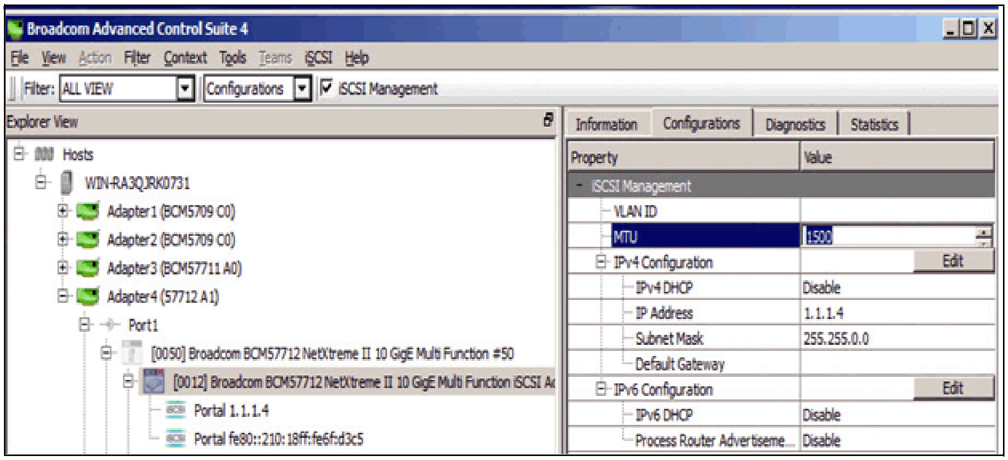
One place to set the Ethernet protocol-enabled partition's adapter MTU size is in the Window's Networking Adapter - Advanced Properties - Jumbo Packet properties.



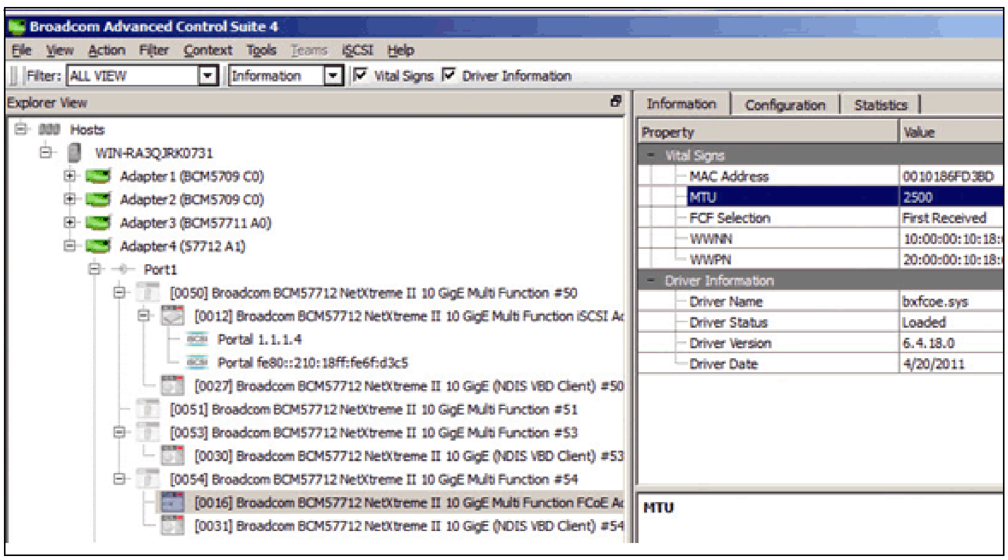
Another place to set the MTU size is in the BACS4 NDIS device Configurations page.



The MTU size for each individual iSCSI Offload (HBA) protocol-enabled partition can be independently set from Normal (1500 bytes) up to Jumbo (9600 bytes) in the BACS4 iSCSI Management Configuration page.



The FCoE device MTU frame size is fixed at 2500 bytes and is not adjustable but is viewable in the FCoE device Information page.



In Windows, each individual partition's Ethernet and iSCSI protocol-enabled adapter MTU size setting can be different. For example:

- Port 0, partition 1 **Ethernet** can be set to 9600 bytes.
- Port 0, partition 1 **iSCSI Offload HBA** can be set to 2500 bytes.
- Port 0, partition 2 **Ethernet** can be set to 1500 bytes.
- Port 0, partition 2 **iSCSI Offload HBA** can be set to 9600 bytes.
- Port 0, partition 3 **Ethernet** can be set to 3500 bytes.
- Port 0, partition 4 **Ethernet** can be set to 5500 bytes.

In Windows, use the ping command with the “-f” option to set the Don't Fragment (DF) flag AND the “-l size” option (small l) to verify that Jumbo Frame support is configured throughout the desired network path - i.e. “ping -f -l 8972 A.B.C.D”. The unfragmentable ping packet size is the desired MTU size to be checked (9000 bytes) minus the automatically added overhead (28 bytes) or 8972 bytes.

```
C:\> ping –f -l 8972 192.168.20.10
Pinging 192.168.20.10 from 192.168.20.50 with 8972 bytes of data:
Reply from 192.168.20.10: bytes=8972 time<1ms TTL=64
Reply from 192.168.20.10: bytes=8972 time<1ms TTL=64
Reply from 192.168.20.10: bytes=8972 time<1ms TTL=64
Reply from 192.168.20.10: bytes=8972 time<1ms TTL=64
Ping statistics for 192.168.20.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

If it does not work, you might see the following reply (if there is connectivity – try 1472 byte standard frames to see if the non-jumbo frame size is passing through):

```
C:\> ping –f -l 8972 192.168.20.10
Pinging 192.168.20.10 from 192.168.20.50 with 8972 bytes of data:
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.
Ping statistics for 192.168.20.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)
```

Setting MTU Sizes in Linux

In Linux, the MTU size for each individual Ethernet protocol-enabled partition can be independently set from Normal (1500 bytes) up to Jumbo (9600 bytes).

Both the Ethernet protocol and iSCSI Offload (HBA) enabled partition’s adapter MTU size is adjusted at the same time using the `ifconfig` command.

```
ifconfig eth3 mtu NNNN up
```

From above, `eth3` is the port identification of the specific 57712-k/578xxS partition to adjust the MTU size. The `NNNN` is the new size of the MTU for that partition, and can be set from 1500 to 9600 bytes.

The following shows all eight partitions being set to different MTU values.

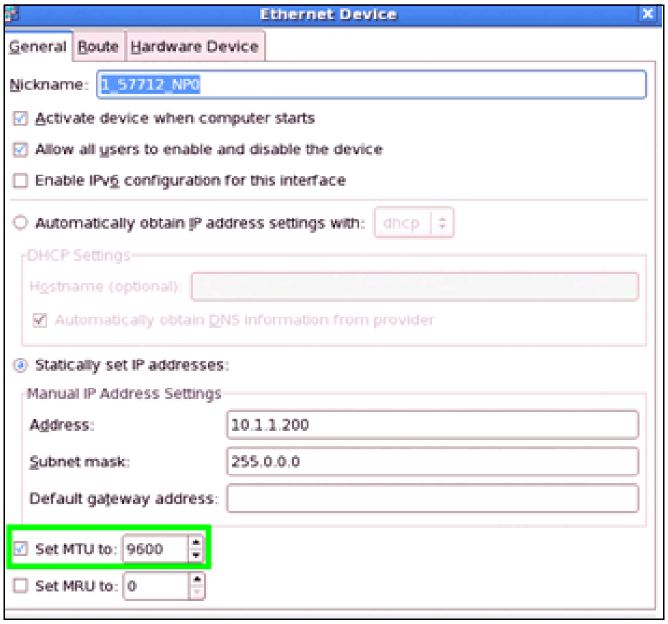
```
[root@localhost ~]#
[root@localhost ~]# ifconfig eth2 mtu 2200 up
[root@localhost ~]# ifconfig eth3 mtu 3300 up
[root@localhost ~]# ifconfig eth4 mtu 4400 up
[root@localhost ~]# ifconfig eth5 mtu 5500 up
[root@localhost ~]# ifconfig eth6 mtu 6600 up
[root@localhost ~]# ifconfig eth7 mtu 7700 up
[root@localhost ~]# ifconfig eth8 mtu 8800 up
[root@localhost ~]# ifconfig eth9 mtu 9600 up
[root@localhost ~]#
```

```
eth2      Link encap:Ethernet  HWaddr 00:10:18:88:E7:A8
          inet addr:10.1.1.200  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7a8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:2200  Metric:1
          RX packets:251 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:47008 (45.9 KiB)  TX bytes:0 (0.0 b)
          Interrupt:169 Memory:d1800000-d1ffffff

eth3      Link encap:Ethernet  HWaddr 00:10:18:88:E7:AA
          inet addr:10.1.1.201  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7aa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:3300  Metric:1
          RX packets:418 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:80164 (78.2 KiB)  TX bytes:0 (0.0 b)
          Interrupt:225 Memory:d2800000-d2ffffff

eth4      Link encap:Ethernet  HWaddr 00:10:18:88:E7:AC
          inet addr:10.1.1.202  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7ac/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:4400  Metric:1
          RX packets:376 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:73184 (71.4 KiB)  TX bytes:0 (0.0 b)
          Interrupt:225 Memory:d3800000-d3ffffff
```

```
eth5      Link encap:Ethernet  HWaddr 00:10:18:88:E7:AE
          inet addr:10.1.1.203  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7ae/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:5500  Metric:1
          RX packets:373 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:72266 (70.5 KiB)  TX bytes:0 (0.0 b)
          Interrupt:204 Memory:d4800000-d4ffffff

eth6      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B0
          inet addr:10.2.2.200  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:6600  Metric:1
          RX packets:371 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:72138 (70.4 KiB)  TX bytes:0 (0.0 b)
          Interrupt:204 Memory:d5800000-d5ffffff

eth7      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B2
          inet addr:10.2.2.201  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:7700  Metric:1
          RX packets:336 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:63944 (62.4 KiB)  TX bytes:0 (0.0 b)
          Interrupt:181 Memory:d6800000-d6ffffff
```

```
eth8      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B4
          inet addr:10.2.2.202  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:8800  Metric:1
          RX packets:316 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:60760 (59.3 KiB)  TX bytes:0 (0.0 b)
          Interrupt:181 Memory:d7800000-d7ffffff

eth9      Link encap:Ethernet  HWaddr 00:10:18:88:E7:B6
          inet addr:10.2.2.203  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::210:18ff:fe88:e7b6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9600  Metric:1
          RX packets:181 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:33702 (32.9 KiB)  TX bytes:0 (0.0 b)
          Interrupt:169 Memory:d8800000-d8ffffff
```

Additionally, the Ethernet and iSCSI protocol-enabled adapter MTU sizes can be simultaneously adjusted in the Ethernet Devices window from the Network Configuration GUI (if available in the desired version of Linux).



Note: In Linux, both the Ethernet and iSCSI Offload HBA MTU sizes are changed simultaneously and will have the same value. In other words, setting eth2 to MTU size of 9600 bytes using the `ifconfig eth2 mtu 9600 up` sets this example's Port 0, Partition 1 Ethernet adapter to MTU = 9600 bytes and the Port 0, Partition 1 iSCSI Offload HBA adapter to MTU = 9600 bytes.

Note: Each partition's MTU size setting can be different. Using the above protocol-enabled partition example:

- Port 0, Partition 1 Ethernet and iSCSI Offload HBA can be set to 9600 bytes.
- Port 0, Partition 2 Ethernet and iSCSI Offload HBA can be set to 5500 bytes.
- Port 0, Partition 3 Ethernet can be set to 1500 bytes.
- Port 0, Partition 4 Ethernet can be set to 9000 bytes.

In Linux, use the ping command with the "-s size" option to verify that Jumbo Frame support is configured throughout the desired network path - i.e. "ping -s 8972 A.B.C.D". The Don't Fragment (a.k.a. DF) flag is AUTOMATICALLY set in Linux. The unfragmentable ping packet size is the desired MTU size to be checked (9000 bytes) minus the automatically added overhead (28 bytes) or 8972 bytes. Note that the ping will return the send size plus the 8 bytes of the ICMP header. The "-c" switch sets the number of times this ping command is sent. You can also use the optional source interface (Capital I) command "-I a.b.c.d" or "-I devName" if you have multiple adapters that are connected to the same target IP address and you want to check a specific interface.

```
[root@server]# ping -c 3 -s 8972 192.168.20.10

PING 192.168.20.10 (192.168.20.10) from 192.168.20.200:8972(9000) bytes of data.

8980 bytes from 192.168.20.10 (192.168.20.10): icmp_seq=0 ttl=255 time=0.185 ms

8980 bytes from 192.168.20.10 (192.168.20.10): icmp_seq=1 ttl=255 time=0.177 ms
```

```
8980 bytes from 192.168.20.10 (192.168.20.10): icmp_seq=2 ttl=255 time=0.180 ms
--- 192.168.20.10 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet lost, time 3043ms
rtt min/ave/max/mdev = 0.177/0.181/0.185/0.005 ms
```

If it does not work, you will see:

```
[root@server]# ping -c 3 -s 8972 192.168.20.10
PING 192.168.20.10 (192.168.20.10) from 192.168.20.200:8972(9000) bytes of data.
--- 192.168.20.10 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet lost, time 3010ms
```

Setting MTU Sizes in VMWare ESX/ESXi 4.1

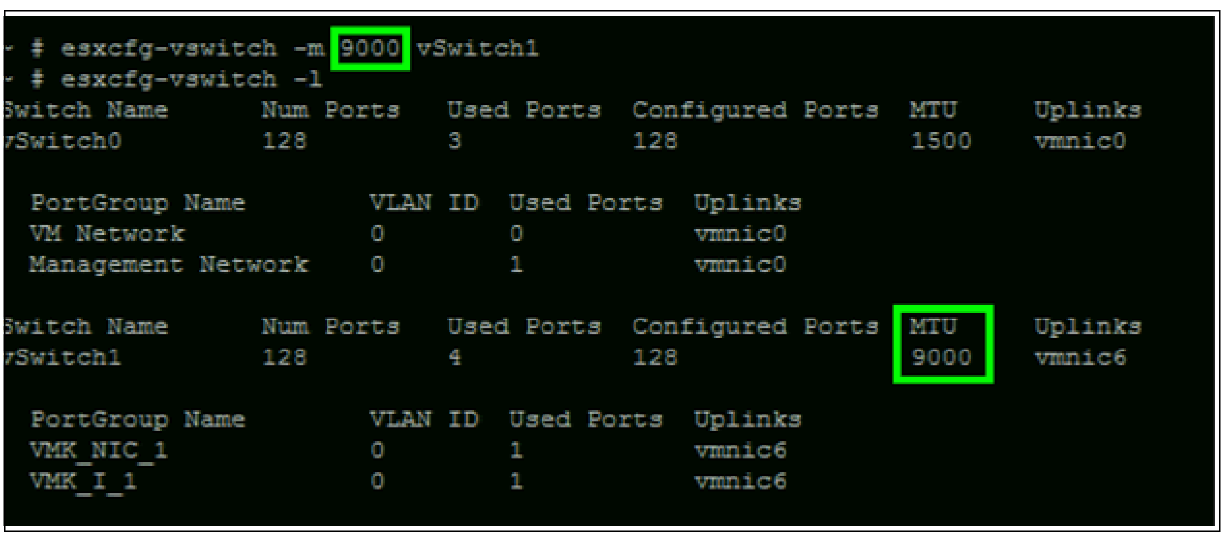
In VMWare ESX/ESXi 4.1, the MTU size for each individual Ethernet protocol-enabled partition can be independently set from Normal (1500 bytes) up to Jumbo (9600 bytes). This MTU size change will affect both regular L2 Ethernet and iSCSI software non-offload pathway generated traffic. Unlike other Linux OS's, you can not directly adjust the MTU size using the “ifconfig” command. Instead you must use various ESX 4.1 “esxcfg” commands.

The Ethernet protocol enabled partition's adapter MTU size is adjusted using SSH command line, for example:

List all current vSwitch's information using the `esxcfg-vswitch -l` command.

Modify a specific vSwitch's MTU size with the following command - this modifies vSwitch1 (which is in port group VMK_I_1) to 9000 bytes:

```
esxcfg-vswitch -m 9000 vSwitch1
```

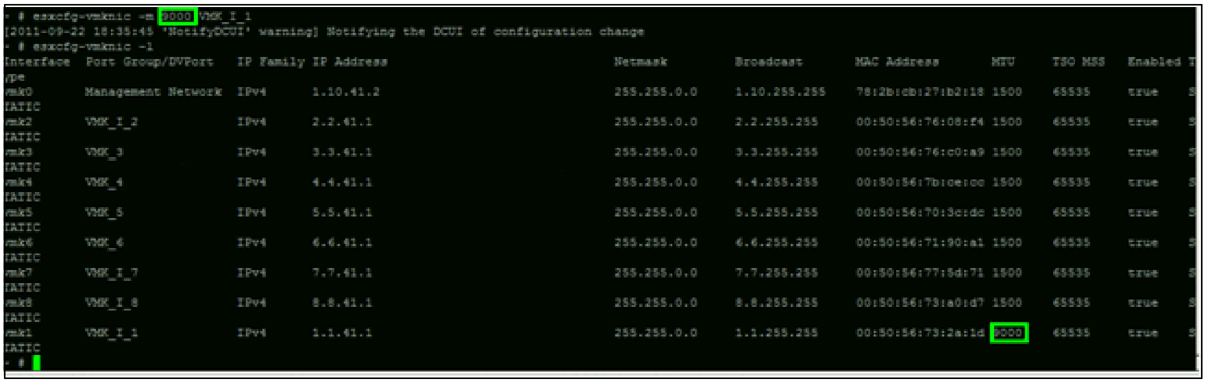


The following command will list all of the current VMKernel NIC settings:

```
esxcfg-vmknic -l
```

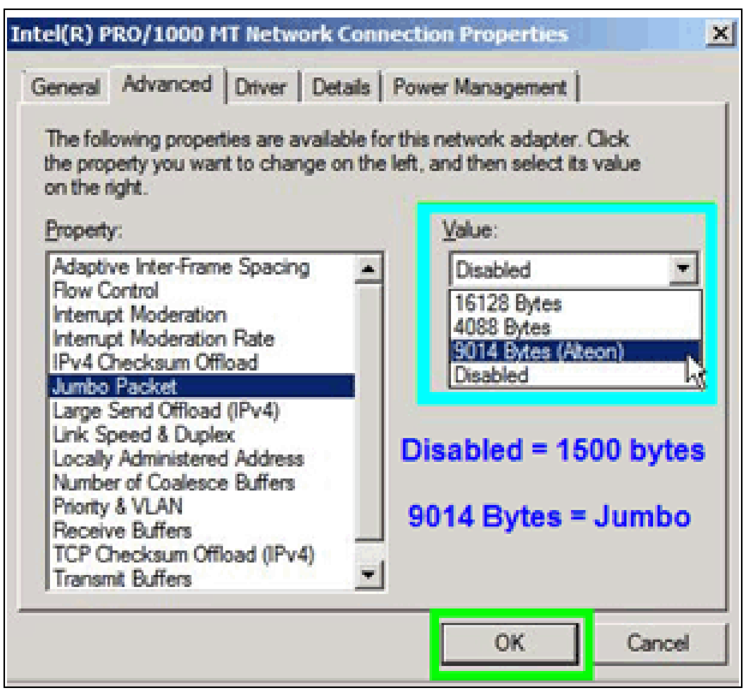
Modify the VMKernel NIC's MTU size using the following command. This modifies **vmnic1** (which is in port group VMK_I_1) to 9000 bytes:

```
esxcfg-vmknic -m 9000 VMK_I_1
```

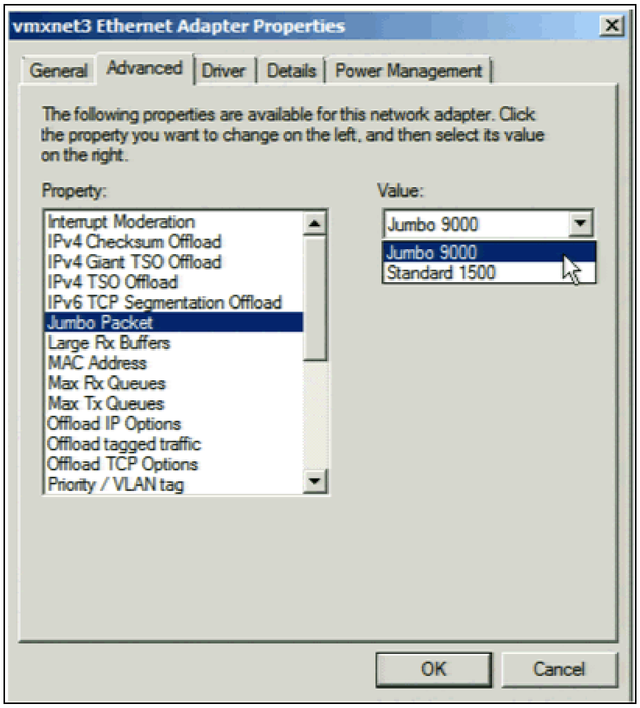


Note: VMWare ESX uses the term **vmnic#** instead of the typical Linux term **eth#** in the command line.

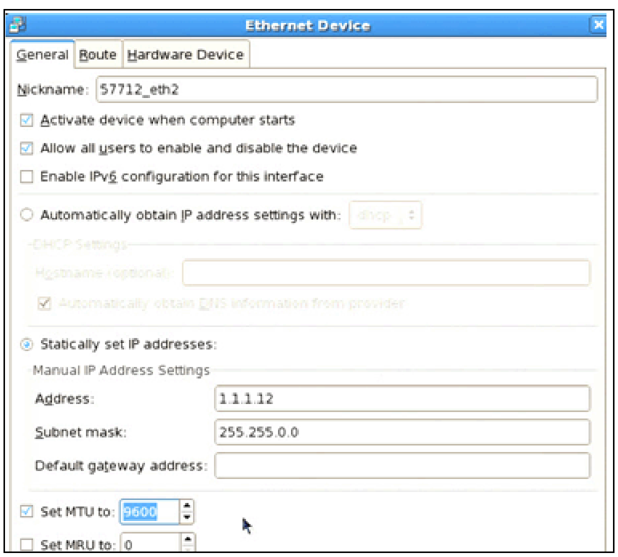
Finally, don't forget to set the associated VM's virtual adapter setting as desired. In a Windows Server 2008 R2 VM, the **Network Connection Advanced Properties Jumbo Packet** setting would be either 9014 Bytes or DISABLED for standard size 1500 Bytes if the Virtual Adapter is the default E1000.



If you are using the VMXNET3 Virtual Adapter, then change it's **Advanced Properties Jumbo Packets** setting to **Jumbo 9000**.



In a RHEL VM, the virtual adapter MTU sizes can be adjusted in the Ethernet Devices window from the **Network Configuration** GUI (or command line using the `ifconfig ethX mtu NNNN up` command).



In Linux, use the following ping command to verify that Jumbo Frame support is configured throughout the desired network path:

Note: The odd numbering for the ping packet size (-s NNNN), which is the desired MTU size of 9000 bytes minus the 28 bytes of automatically added OVERHEAD, but the outputted displayed size is plus 8 bytes which are the included ICMP header data bytes. The Don't Fragment (a.k.a. DF) flag (-d) must also be set, else the packet could be fragmented somewhere along the way and you would not know it. The optional count (-c) is the number of times you will send this ping. You don't have to use the optional source interface command "-I a.b.c.d" or "-I devName" unless you have multiple adapters that lead to the same target IP address and you want to check a specific one. You can also use the ESX "vmkping" command.

```
[root@sieora1 network-scripts]# ping -d -c 3 -I eth1 -s 8972 192.168.20.10
PING 192.168.20.10 (192.168.20.10) from 192.168.20.101 eth1:8972(9000) bytes of data.
8980 bytes from 192.168.20.10: icmp_seq=1 ttl=255 time=0.185 ms
8980 bytes from 192.168.20.10: icmp_seq=2 ttl=255 time=0.177 ms
8980 bytes from 192.168.20.10: icmp_seq=2 ttl=255 time=0.180 ms
--- 192.168.20.10 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet lost
round-trip min/ave/max = 0.177/0.181/0.185ms
```

If it does not work you will see a failure message:

```
[root@esx41]# ping –d –c 3 -I eth1 -s 8972 192.168.20.10
PING 192.168.20.10 (192.168.20.10) from 192.168.20.101 eth1:8972(9000) bytes of data.
sendto () failed (Message too long)
sendto () failed (Message too long)
sendto () failed (Message too long)

--- 192.168.20.10 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet lost
```

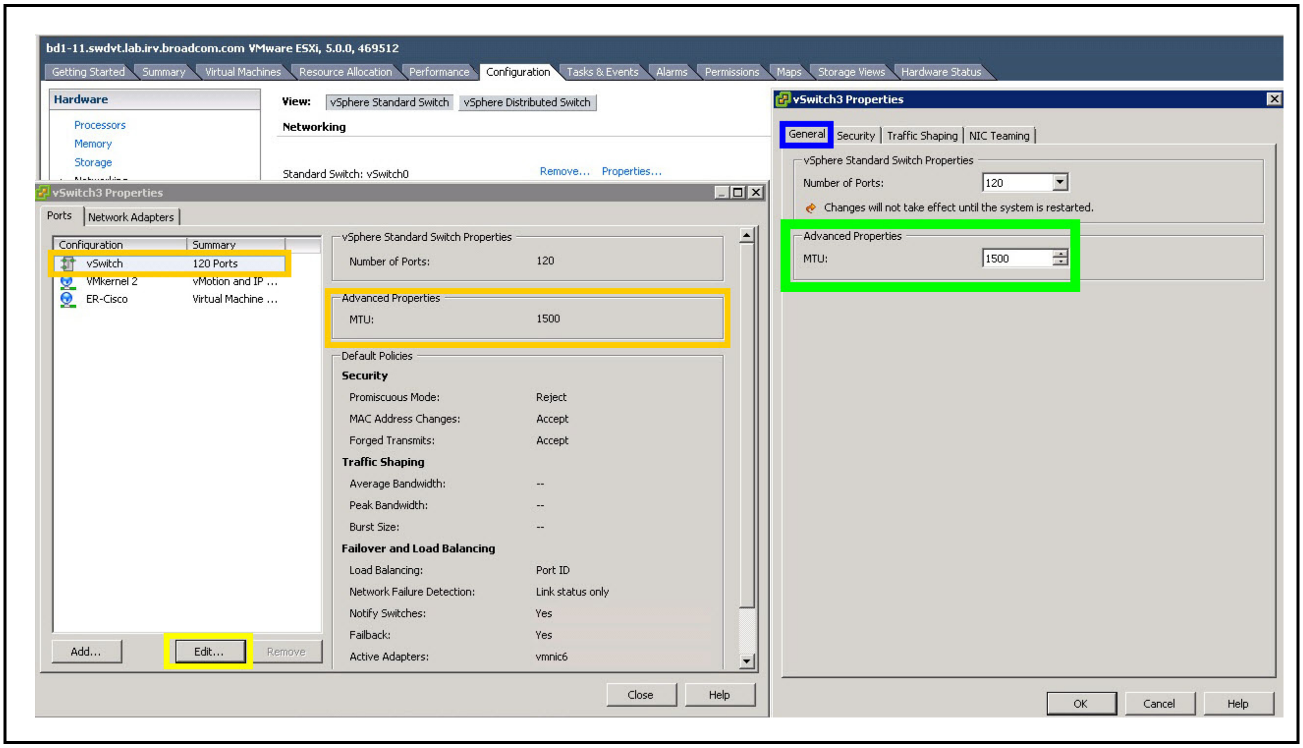
You should also check if the desired VMs can send Jumbo Frames all the way to their various end points using the applicable OS ping commands described earlier in this document.

Setting MTU Sizes in VMWare ESXi 5.0

In VMWare ESXi 5.0, the MTU size for each individual Ethernet protocol-enabled partition can be independently set from Normal (1500 bytes) up to Jumbo (9600 bytes). This MTU size change affects both regular L2 Ethernet and iSCSI software non-offload pathway generated traffic. Unlike other Linux OS's, you can not directly adjust the MTU size using the "ifconfig" command. Instead, you can use various ESX 4.1 "esxcfg" commands or change them via ESXi 5.0's vSphere.

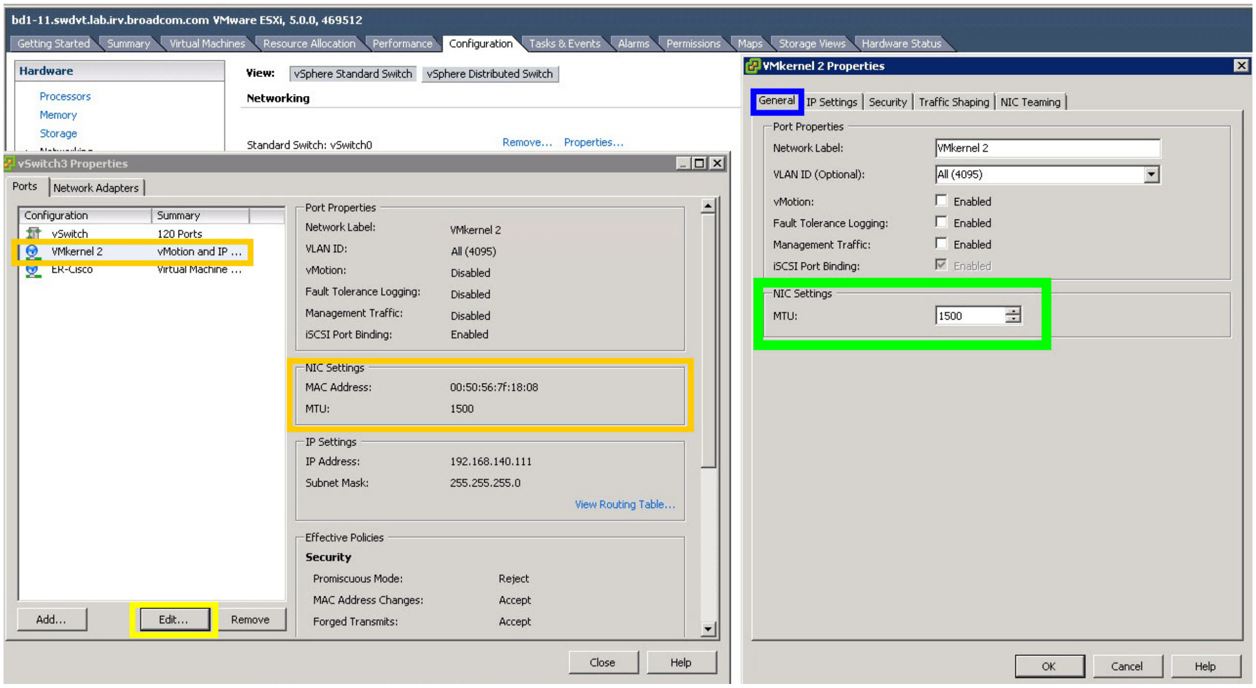
1. In ESXi 5.0's vSphere, enter the desired vSwitch's Properties and select **vSwitch** in the Ports Tab. This displays the current MTU size setting for that vSwitch.

   To change the size, click the Edit button under the Ports tab. In the new window's General tab, adjust the Advanced Properties MTU value as desired and click **OK**.



2. In the same vSwitch's Properties widow, select **VMKernel** in the Ports Tab. This displays the current NIC MTU size setting for that VMKernel of that vSwitch.

   To change the size, click the same **Edit** button under the Ports tab. In the new window's General tab, adjust the NIC Settings MTU value as desired.



The Ethernet protocol enabled partition's adapter MTU size can also be adjusted using the same commands previously described for ESX/ESXi 4.1. Be sure to set the associated VM's virtual adapter setting as desired and to use ping's to ensure the desired MTU size (without switch fragmentation) is supported throughout your network.

Examples

- Equal Oversubscription Example
- Partitioned Oversubscription Example
- Weighted Oversubscription Example
- Oversubscription With One High Priority Partition Example
- Default Fixed Subscription Example
- Mixed Fixed Subscription and Oversubscription Example
- Mixed Weights and Subscriptions Example

Note: All bandwidths given in these examples are approximations. Protocol overhead, application send-rate variances, and other system limitations may give different bandwidth values, but the ratio relationship between the send bandwidths of the four partitions on the same port should be similar to the ones given.

Depending upon OS requirements, on the 57712-k/57810S the traffic types for each partition could be L2 Ethernet or iSCSI Offload or FCoE Offload where L2 Ethernet can be on any of the four partitions, AND up to two iSCSI Offload can be on any two of the four partitions OR one FCoE Offload can be on any one of the four partitions plus one iSCSI Offload can be on any one of the remaining three partitions. The partitions data flows on Windows can be a combination of L2 Ethernet traffic (with or without TOE) and/or HBA traffic (iSCSI OR FCoE). For VMWare ESX/ESXi 4.1 and Linux RHEL v5.x, the L2 Ethernet protocol is always enabled and up to two iSCSI Offloads can be enabled (but no FCoE Offload). For Linux RHEL v6.x and SLES 11 SP1 the L2 Ethernet protocol is always enabled and up to two iSCSI Offloads or one FCoE Offload and one iSCSI Offload can be enabled. For Citrix® XenServer 6.0 only the L2 Ethernet protocol is available for the partitions. For Oracle Solaris® 10, the L2 Ethernet protocols and up to one FCoE Offload can be enabled. For VMWare ESXi 5.0, the L2 Ethernet protocols are also always enabled and up to two iSCSI Offloads or one FCoE Offload and one iSCSI Offload can be enabled.

Equal Oversubscription Example

The following is an example of oversubscribed bandwidth sharing in non-DCB mode. All traffic types over the four partitions of the port have an equal weight (i.e., they are all set to 0%) and can individually use the maximum bandwidth of the connection (i.e., 10 Gbps, in this case). In addition to the Ethernet Protocol's being enabled on all four partitions, the iSCSI Offload protocol is enabled on partition 1 and 4. The iSCSI Offload Protocol can be enabled in any two partitions. When all of the partitions have zero relative bandwidth weights, each traffic flow will act as if in it's own separate partition, each taking an equal share of the available bandwidth up to that partitions maximum bandwidth (which is 100% in this example so does not further limit any of the traffic flows).

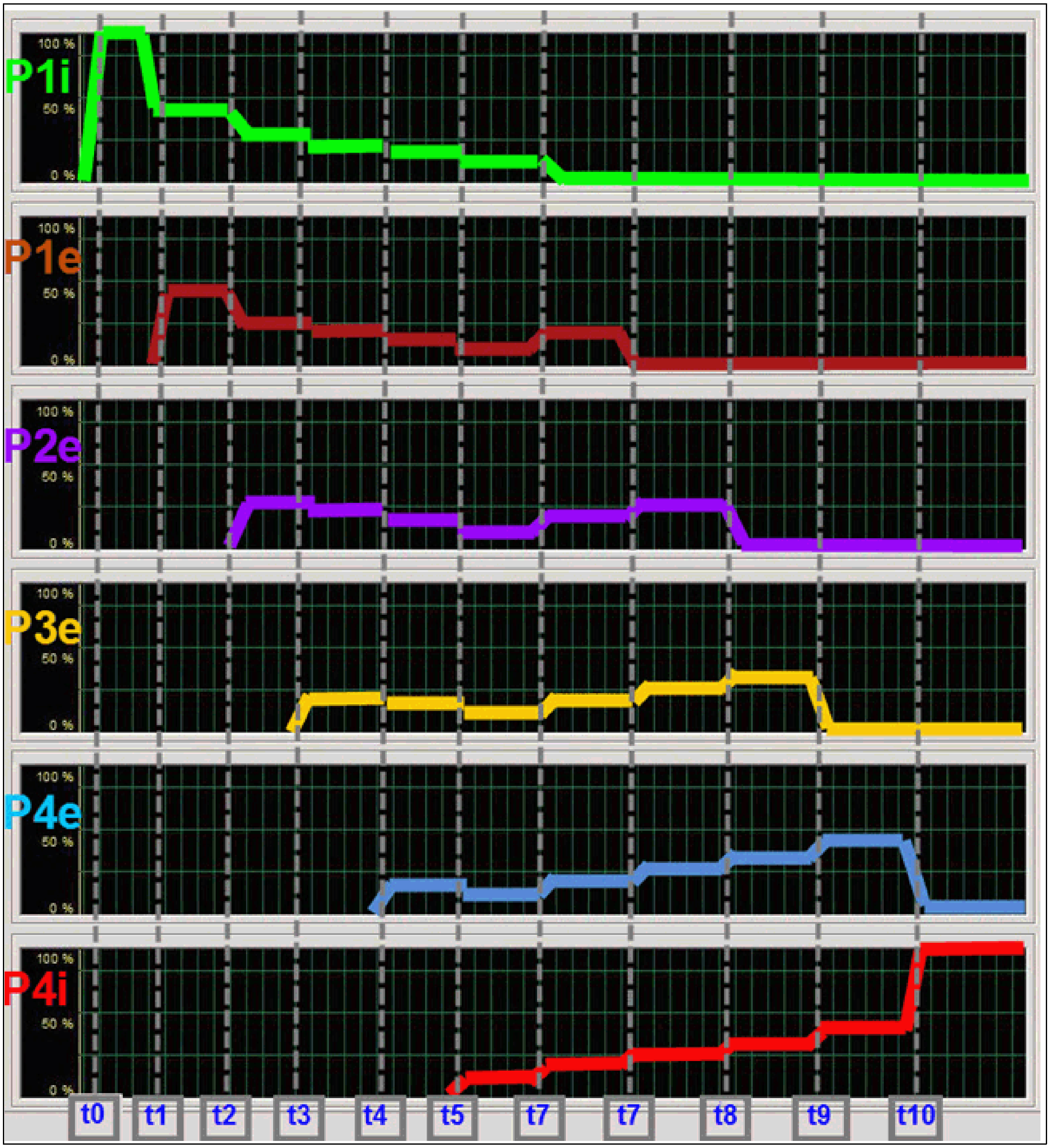
FCoE Offload is not available in non-DCB mode so two iSCSI Offload protocols are used here.

Table 4: Non-DCB Equal Oversubscription

Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	0	100	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	0	100	Ethernet	Brown
Port 0, Partition 2 (P2e)	0	100	Ethernet with TOE	Purple
Port 0, Partition 3 (P3e)	0	100	Ethernet	Yellow
Port 0, Partition 3 (P4e)	0	100	Ethernet with TOE	Blue
Port 0, Partition 4 (P4i)	0	100	iSCSI Offload	Red

The following plot shows how all of the partitions would share a ports available send bandwidth. Traffic flows on the same partition (such as P1i and P1e) are expanded in to their own bandwidth trace for ease of understanding. The send traffic flows are independent in each partition and the individual traffic type flow rate is balanced with each of the other traffic type flow rates when traffic demands exceed the available bandwidth

- Starting at **t0**, the first partition's iSCSI Offload traffic flow (P1i) initially takes ~100% of the available port's TX send bandwidth when an iSCSI test application is flooding that port by itself.
- When P1's L2 Ethernet (P1e) starts to send at **t1**, both stabilize to half of the bandwidth or ~5 Gbps each even thought they are in the same partition, they share the total available bandwidth.
- When P2e starts to send at **t2**, all three traffic flows (P1i, P1e and P2e) will stabilize to 1/3rd of the bandwidth or ~3.3 Gbps each (they all equally share the available bandwidth).
- When P3e starts to send at **t3**, all four traffic flows (P1i, P1e, P2e and P3e) will stabilize to 1/4th of the bandwidth or ~2.5 Gbps each (effectively sharing the available bandwidth).
- When P4e starts to send at **t4**, all five traffic flows (P1i, P1e, P2e, P3e and P4e) will stabilize to 1/5th of the bandwidth or ~2 Gbps each (again equally sharing the available bandwidth).
- When P4i starts to send at **t5**, all six traffic flows (P1i, P1e, P2e, P3e, P4e and P4i) will stabilize to 1/6th of the bandwidth or ~1.65 Gbps each (all sharing the available bandwidth).
- When P1i stops sending at **t6**, the five currently active traffic flows (P1e, P2e, P3e, P4e and P4i) will readjust to ~2 Gbps each (equally absorbing the freed up bandwidth).
- As a previously sending traffic flow stops sending (**t7**, **t8**, **t9** and **t10**) the remaining active flows will readjust to equally fill any available bandwidth.
- Notice the symmetry of the BW allocation. No matter which traffic type is currently running, each will get an equal share with respect to the other currently transmitting traffic type flows. This assumes the application creating the transmitted traffic type flow can fill the allocated amount of BW it is given - if not, the other traffic flows will equally absorb the unused BW.


P1i
P1e
P2e
P3e
P4e
P4i
100 %
50 %
0 %
t0
t1
t2
t3
t4
t5
t7
t7
t8
t9
t10

The following two examples are of oversubscribed bandwidth sharing with DCB enabled. The first example is similar to the above non-DCB example with four Ethernet and two iSCSI Offload protocols enabled with the same two traffic types in six distinct “flows” with the partitions similarly configured. The main differences here is the iSCSI Offloaded traffic type is assigned to DCB Priority Group 2 and is Lossless (i.e. iSCSI-TLV) with an ETS setting of 50%; while the L2 Ethernet traffic type is still assigned to Priority Group 0 and is Lossy with an ETS setting of 50%. If the iSCSI Offload protocol traffic flows had been assigned to the same PG as the Ethernet protocol traffic flows, then the traffic BW would have looked very similar to the previous Non-DCB example since ETS would never be activated for traffic flows belonging to the same PG.

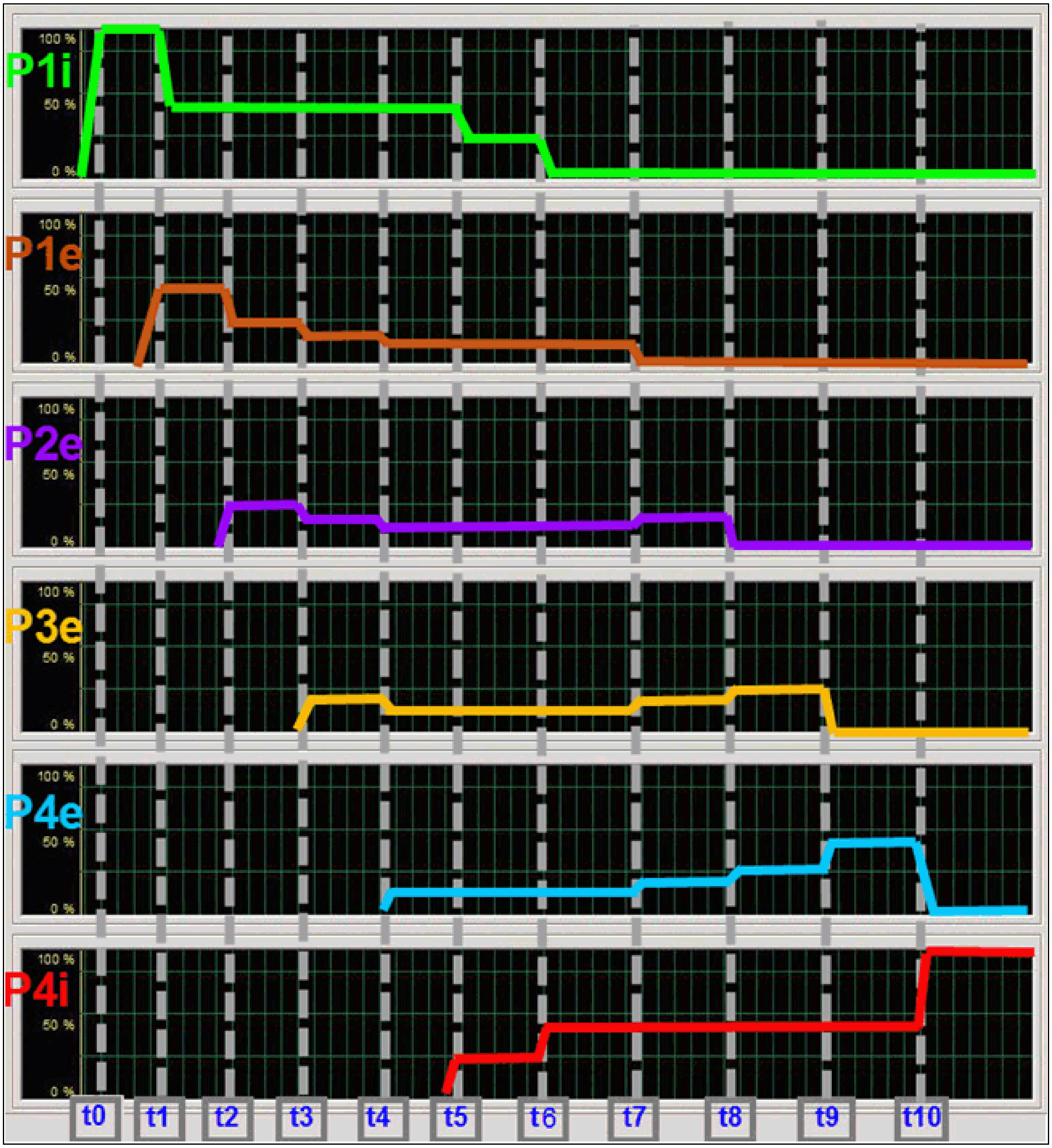
Table 5: DCB Equal Oversubscription

Port, Partition	***Relative Bandwidth Weight (%)***	***Maximum Bandwidth***	***Protocol***	***Plot Color***
Port 0, Partition 1 (P1i)	N/A	100	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	N/A	100	Ethernet	Brown
Port 0, Partition 2 (P2e)	N/A	100	Ethernet with TOE	Purple
Port 0, Partition 3 (P3e)	N/A	100	Ethernet	Yellow
Port 0, Partition 4 (P4e)	N/A	100	Ethernet	Blue
Port 0, Partition 4 (P4i)	N/A	100	iSCSI Offload	Blue

The following plot shows how the two iSCSI traffic streams in PG2 act verses the L2 Ethernet traffic streams in PG0. The traffic in the two PGs will almost act independently of each other when their aggregated traffic bandwidth demands exceed the available bandwidth - each taking it's half of the ETS managed pie.

- Starting at **t0**, only P1i (iSCSI Offload) is sending, so it takes ~100% or all of the 10 Gbps bandwidth.
- When P1e (L2 Ethernet) starts to send at **t1**, both flows stabilize to ~5 Gbps each (P1i in PG2 takes it's allocated 50% bandwidth and P1e in PG0 takes it's allocated bandwidth of 50%).
- When P2e starts to send at **t2**, the traffic in P1i is not affected - it remains at ~5 Gbps due to it being in a different Priority Group. Both P1e and P2e will stabilize to ~2.5 Gbps each (P1e and P2e equally share PG0's allocated portion of the bandwidth).
- When P3e starts to send at **t3**, P1i is still unaffected and remains at ~5 Gbps. The three L2 Ethernet traffic types will split their 50% of PG0's share between themselves, which is ~1.65 Gbps (each takes 1/3rd of ~5 Gbps).
- When P4e starts to send at **t4**, the four Ethernet traffic flows take 1/4th of PG0's bandwidth (~1.25 Gbps each) while P1i is still unaffected and remains at ~5 Gbps.
- When P4i starts to send at **t5**, the four Ethernet traffic flows remain the same but the two iSCSI traffic flows split PG2's allocated bandwidth (~2.5 Gbps each).
- Then when P1i stops sending at **t6**, the traffic flows in PG0 are unaffected while P4i's share increases to all of PG2's allocated bandwidth of ~5 Gbps.
- As each of the traffic flows stops sending in PG0, the traffic flows of the remaining member's of PG0 equally increase their respective shares to automatically occupy all of the available bandwidth remain in PG0 until all of PG0's Ethernet flows stop. At **t7**, there are three active PG0 flows (P2e, P3e and P4e) so each gets 1/3rd of PG0's 5 Gbps or ~1.65 Gbps. At **t8**, there are two active PG0 flows (P3e and P4e) so each gets ½ of PG0's 5 Gbps or ~2.5 Gbps. At **t9**, there is only one active PG0 flow (P4e) so it gets all of PG0's bandwidth or ~5 Gbps. Through all of this, the Lossless iSCSI flow of P4i remains at 5 Gbps since it takes all

of PG2's portion of the overall bandwidth (ETS of 50%). Finally, at **t10**, there is only one active flow after P4e stops sending so at this point P4i gets 100% of all the bandwidth or ~10 Gbps.



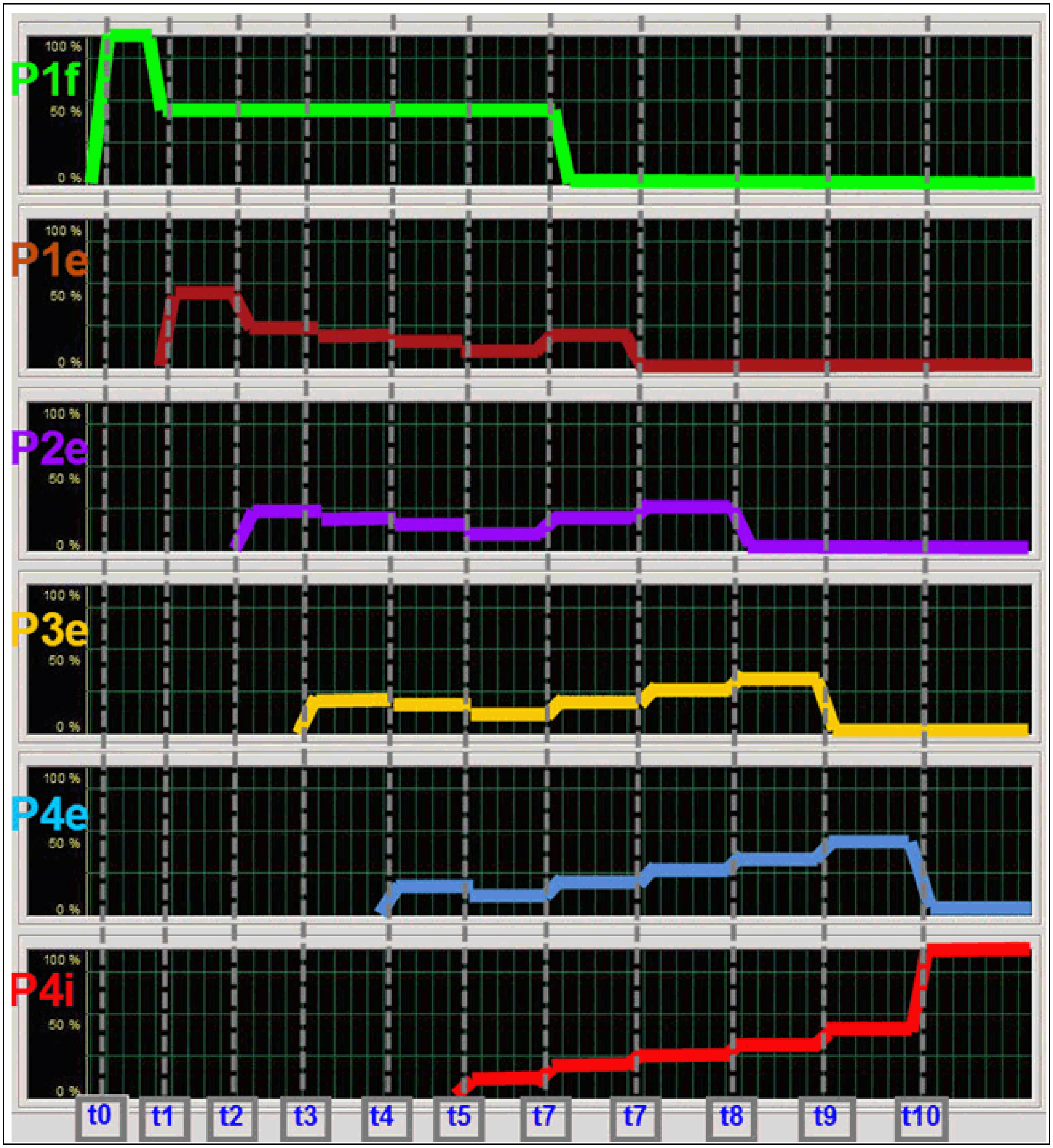
This second DCB example of oversubscribed bandwidth sharing replaces one of the iSCSI Offloads with an FCoE Offload protocol. This gives a total of three distinct traffic types in six “flows” with similar partition settings. The other difference is that the FCoE traffic type is assigned to DCB Priority Group 1 and is Lossless with an ETS setting of 50%; while the L2 Ethernet and iSCSI Hardware Offload traffic types are both now in Priority Group 0 and are both Lossy with an ETS setting of 50%.

Table 6: DCB Equal Oversubscription with one Lossless FCoE Offload

Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1f)	N/A	100	FCoE Offload	Green
Port 0, Partition 1 (P1e)	N/A	100	Ethernet with TOE	Brown
Port 0, Partition 2 (P2e)	N/A	100	Ethernet	Purple
Port 0, Partition 3 (P3e)	N/A	100	Ethernet	Yellow
Port 0, Partition 4 (P4e)	N/A	100	Ethernet with TOE	Blue
Port 0, Partition 4 (P4i)	N/A	100	iSCSI Offload	Red

The following plot shows how the first partition's FCoE traffic PG1 acts verses the other traffic types in PG0. Just like the previous example, the traffic in the two PGs will almost act independently of each other when their aggregated traffic bandwidth demands exceed the available bandwidth.

- Starting at **t0**, only P1f (FCoE Offload) is sending, so it takes ~100% or all of the 10 Gbps bandwidth.
- When P1e (L2 Ethernet) starts to send at **t1**, both flows stabilize to ~5 Gbps each (P1f in PG1 takes 50% and P1e in PG0 takes the other 50%).
- When P2e starts to send at **t2**, the traffic in P1f is not affected - it remains at ~5 Gbps due to it being in a different PG. Both P1e and P2e will stabilize to ~2.5 Gbps each (P1e and P2e equally share PG0's portion of the bandwidth - so they each get ~2.5 Gbps (ETS of 50% * total 10G BW * 1 / 2).
- When P3e starts to send at **t3**, P1f is still unaffected and remains at ~5 Gbps. The three L2 Ethernet traffic types will split their 50% of PG0's share between themselves which is ~1.65 Gbps (each takes 1/3rd of 5G).
- When P4e starts to send at **t4**, the four Ethernet traffic flows take 1/4th of PG0's bandwidth (~1.25 Gbps each) while P1f is still unaffected and remains at 5 Gbps.
- When P4i starts to send at **t5**, the four Ethernet traffic flows plus the new iSCSI traffic flow take 1/5th of PG0's bandwidth (~1 Gbps each) while P1f is still unaffected and remains at 5 Gbps.
- Then when P1f stops sending at **t6**, the five traffic flows in PG0 now take all of the ports bandwidth, so now their 1/5th of PG0's bandwidth doubles to ~2 Gbps each - the available bandwidth went from 5 Gbps to 10 Gbps.
- As each traffic flow stops sending in PG0, the remaining member traffic flows equally increase their respective shares to automatically occupy all of the available bandwidth. At **t7**, there are four active PG0 flows so each gets 1/4th or ~2.5 Gbps. At **t8**, there are three active PG0 flows so each gets 1/3rd or ~3.3 Gbps. At **t9**, there are two active PG0 flows so each gets half or ~5 Gbps. Finally, at **t10**, there is only one active PG0 flow (P4i) so it gets 100% or ~10 Gbps.
- Any of the traffic flows will take 100% of the available bandwidth if it is the only sending traffic flow.


P1f
P1e
P2e
P3e
P4e
P4i
100 %
50 %
0 %
t0
t1
t2
t3
t4
t5
t7
t7
t8
t9
t10

Partitioned Oversubscription Example

The following is an example of oversubscribed bandwidth sharing in non-DCB mode where all four partitions of the port have their weight set to 25% and can individually use the maximum bandwidth of the connection (i.e., 10 Gbps, in this case). In addition to the Ethernet Protocol's being enabled on all four partitions, the iSCSI Offload protocol is enabled on Partition 1 and 4. By setting the partition's relative bandwidth weights to 25%, each partition's traffic flows (i.e. P1's iSCSI (P1i) + L2 Ethernet (P1e) and P4's iSCSI (P4i) + L2 Ethernet (P4e)) will be contained in their respective partition while each partition over all takes an equal share of the available bandwidth. The traffic flows within that partition can only expand into that partition's allocated by weight portion.

There would be no difference between the previous examples and this example with DCB enabled since it essentially sets the Relative Bandwidth Weight values to all ZEROs and the PG's ETS would come into play.

Table 7: Non-DCB Partitioned Oversubscription

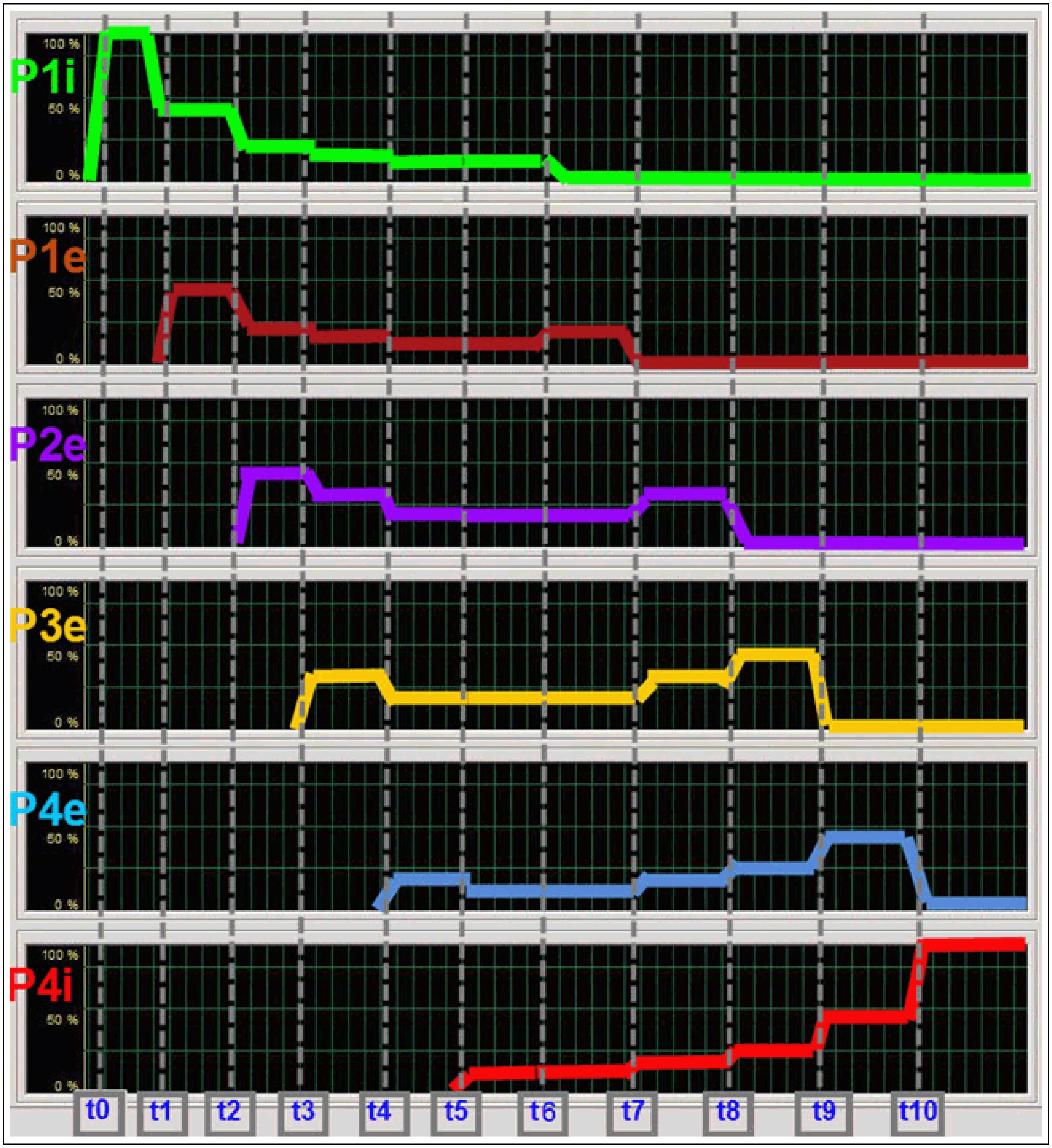
Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	25	100	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	25	100	Ethernet	Brown
Port 0, Partition 2 (P2e)	25	100	Ethernet with TOE	Purple
Port 0, Partition 3 (P3e)	25	100	Ethernet	Yellow
Port 0, Partition 3 (P4e)	25	100	Ethernet with TOE	Blue
Port 0, Partition 4 (P4i)	25	100	iSCSI Offload	Red

The following plot shows how each traffic type flow must remain within a partition's share of a ports available send bandwidth - i.e. if there are two different traffic type flows (such as P1i and P1e) in a single partition, they are combined, as if one flow, for determining the amount of bandwidth allocated to them.

- Starting at **t0**, the first partition's iSCSI Offload traffic flow (P1i) initially takes ~100% of the available port's TX send bandwidth when an iSCSI test application is flooding that port by itself.
- When P1's L2 Ethernet (P1e) starts to send at **t1**, both stabilize to ~5 Gbps each even thought they are in the same partition, they share the total available bandwidth. This is because no other partition's traffic flow is sending.
- When P2e starts to send at **t2**, the traffic flows in P1 (P1i and P1e) will reduce to ~2.5 Gbps each while the P2 traffic flow (P2e) will take ~5 Gbps. This is because the bandwidth is initially split by partition and then traffic flows within each individual partition.
- When P3e starts to send at **t3**, the two traffic flows in P1 (P1i and P1e) are further reduced to ~1.65 Gbps (half of the partition's 1/3rd allocation going to the three active partitions) while P2e and P3e each stabilize at ~3.3 Gbps.
- When P4e starts to send at **t4**, the three single partition traffic flows (P2e, P3e and P4e) will stabilize to 1/4th of 10 Gbps or ~2.5 Gbps each while the P1 partition shares it allocated bandwidth between it's two users (P1i and P1e) so each gets half of the allocated 1/4th of 10 Gbps or 1/8th which is ~1.25 Gbps.
- When P4i starts to send at **t5**, the two single traffic flows (P2e and P3e) will remain at ~2.5 Gbps each, as well as the P1 partition's traffic flows (P1i and P1e) each still getting ~1.25 Gbps while P4's allocated

bandwidth will now be split into two traffic flows (P4e and P4i) which means each get ~1.25 Gbps.

- When P1i stops sending at **t6**, the only partition P1 traffic flow (P1e) will readjust to ~2.5 Gbps and all of the others will remain the same.
- When P1e stops sending at **t7**, the other traffic flows will readjust to ~3.3 Gbps (P2e and P3e) and ~1.65 Gbps each for partition P4's shared P4e and P4i traffic flows.
- When P2e stops sending at **t8**, partition P3's single traffic flow will readjust to ~5 Gbps (P3e) and partition P4's shared P4e and P4i traffic flows will increase to half that or ~2.5 Gbps.
- When P3e stops sending at **t9**, the remaining partition (P4) will now receive all of the available bandwidth so it's two traffic flows (P4e and P4i) will equally share it for ~5 Gbps each.
- Finally, when P4e stops sending at t10, the remaining traffic flow (P4i) will now receive all of the available bandwidth or ~10 Gbps.
- If there is only one flow in a partition, it would take all of the bandwidth allocated for that partition.
- The main difference between setting all four partition's relative bandwidth weight to 0% and setting them to all 25% is that 0%'s causes the send bandwidth to be shared between all active traffic flows while 25%'s cause the send bandwidth to be shared between the active sending partitions first and then the active sending traffic type flows - in a two step manner. Setting them to all 0%'s causes the logic to work similarly to the way it does in DCB mode when all traffic types are in the same PG.
- If there was only one traffic flow in each partition, then the results would be similar to setting each partition's relative bandwidth weight to 0%, since the single traffic flow would not be sharing a partition's bandwidth with another traffic type.


P1i
P1e
P2e
P3e
P4e
P4i
100 %
50 %
0 %
t0
t1
t2
t3
t4
t5
t6
t7
t8
t9
t10

Weighted Oversubscription Example

The following is an example of weighted oversubscribed bandwidth sharing with different weights assigned to each partition in non-DCB mode. This example has each partition taking the maximum bandwidth when no other partition is active, plus as each partition starts and stops sending, the amount of bandwidth is shared as an approximate ratio of the currently sending partitions weight values.

There would be no difference between the previous examples and this example with DCB enabled since the weights are ignored and if all traffic types are in the same single PG.

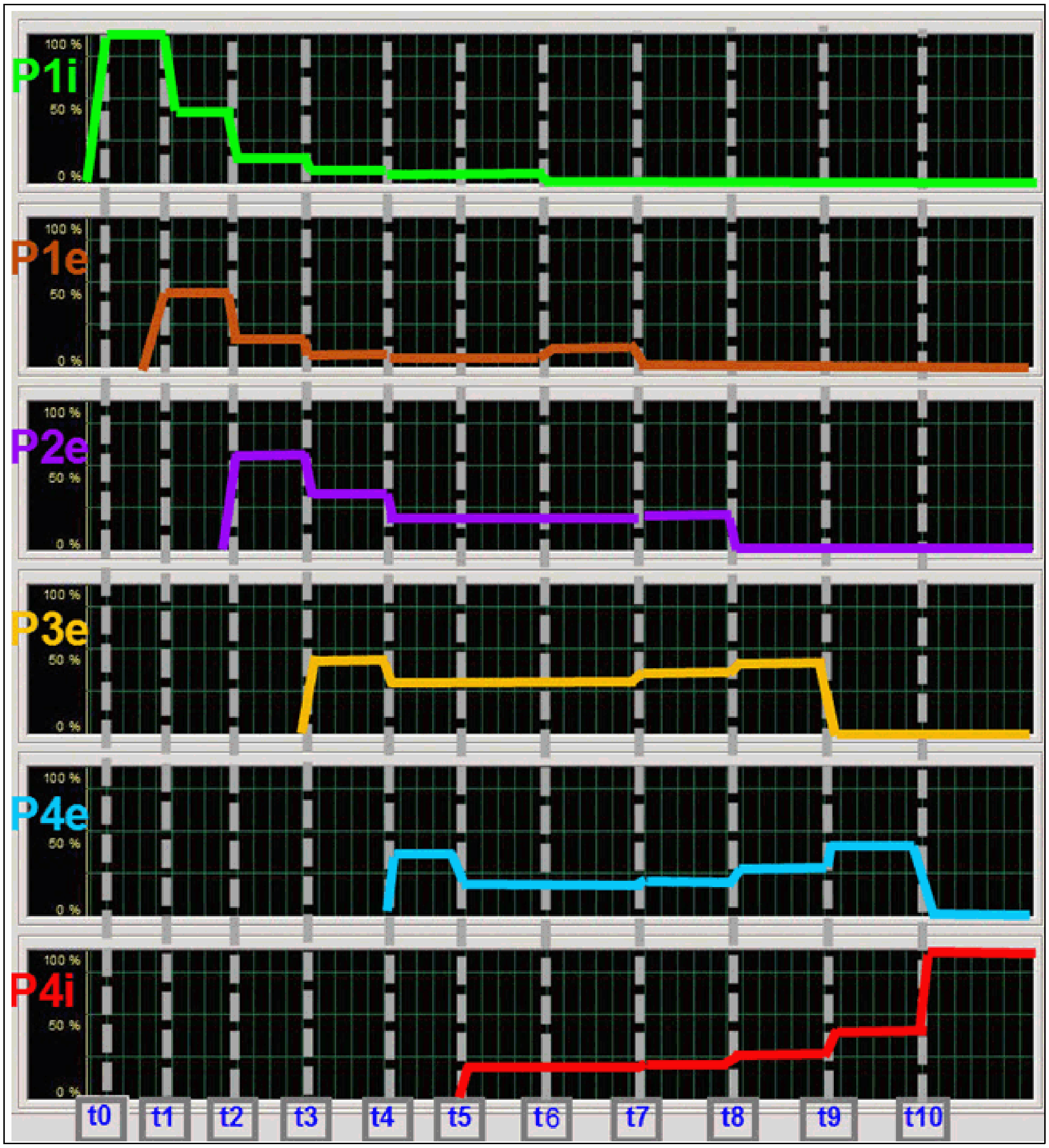
Table 8: Non-DCB Weighted Oversubscription

Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	10	100	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	10	100	Ethernet	Orange
Port 0, Partition 2 (P2e)	20	100	Ethernet	Purple
Port 0, Partition 3 (P3e)	30	100	Ethernet with TOE	Yellow
Port 0, Partition 4 (P4e)	40	100	Ethernet with TOE	Blue
Port 0, Partition 4 (P4i)	40	100	iSCSI Offload	Red

The following plot shows:

- The first partition's traffic flow (P1i) initially takes ~100% of the available bandwidth at **t0** when an iSCSI test application is sending traffic out that port by itself.
- When P1e starts to send Ethernet traffic at **t1**, the two active traffic flows have equal weights with respect to each other so they are allocate half of the total bandwidth available (~10 Gbps) to partition P1 which equates to ~5 Gbps each for P1i and P1e.
- When P2e starts sending at **t2**, the partition's relative bandwidth weights come into effect. Partition P1 has a weight of 10% while P2 has twice as much at 20%, so P1's two sending traffic flows are reduced to half of the partition's assigned 1/3rd (derived from P1's weight of 10% / (P1's weight of 10% + P2's weight of 20%)) or ~1.65 Gbps each for P1i and P1e. P2e starts at ~6.7 Gbps (it's relative weight is 20% / 30% total active weights) - it is not halved since it is the only traffic-flow on partition P2.
- When partition P3e starts sending Ethernet traffic at **t3** with a relative weight of 30%, it takes ~5 Gbps (30/60 of 10 Gbps), P2e drops to ~3.3 Gbps (20/60) and partition P1's total drops to ~1.65 Gbps (10/60) so that means P1i and P1e each get half of that or ~0.825 Gbps each.
- When P4e starts (40% relative weight) at **t4**, it takes ~4 Gbps (40/100) and the three other partition's send traffic drop; partition P1 is reduced to ~1 Gbps (10/100) so that means P1i and P1e split that for ~0.5 Gbps; partition P2 drops to ~2 Gbps (20/100) and since there is only one send traffic flow (P2e) it takes all of that assigned bandwidth; and finally partition P3 (with it's single traffic flow P3e) drops to ~3 Gbps (30/100).
- When the second traffic flow on partition P4 (P4i) starts at **t5**, the two flows (P4e and P4i) on the same partition (P4) split the partition's assigned bandwidth of ~4 Gbps, so each gets ~2 Gbps. The other send traffic on the other three partitions remains the same.
- If P1i stops at **t6**, the remaining traffic flow on partition P1 (P1e) absorbs that partition's share of the send bandwidth to ~1 Gbps. The remaining traffic flows on the other three partitions are unaffected.

- If P1e stops at **t7**, the others adjust slightly upwards to fill that newly available bandwidth; partition P2 (and it's traffic flow P2e) increases to ~2.2 Gbps (20/90); partition P3 (and it's traffic flow P3e) raises to ~3.3 Gbps (30/90); and partition P4 raises to ~4.5 Gbps (40/90) which means P4e and P4i split that for ~2.25 Gbps each.
- If P2e stops at **t8**, the others again adjust upwards to fill that newly available bandwidth; partition P3 (and it's traffic flow P3e) raises to ~4.3 Gbps (30/70), and partition P4 raises to ~5.7 Gbps (40/70) which means P4e and P4i split that for ~2.85 Gbps each.
- If P3e stops at **t9**, partition P4's share raises to ~100% of the bandwidth (40/40) so it's two traffic flows (P4e and P4i) split this for ~5 Gbps each.
- When P4e stops at **t10**, the only remaining traffic flow P4i takes all of the bandwidth at ~10 Gbps.



Oversubscription With One High Priority Partition Example

Next is an example of a single high priority partition with almost all the relative bandwidth weight, but all four of the partitions are still oversubscribing the available bandwidth. The first three partitions of the port have 1% weight and the last partition (P4) has almost all the weight (97%). All four partitions are set to use the maximum amount of the connection's bandwidth (i.e., 100%, which is 10 Gbps, in this case).

Again, since the maximum bandwidths are set to 100% for all four partitions, there is no difference between the earlier DCB mode example and this one, in DCB mode.

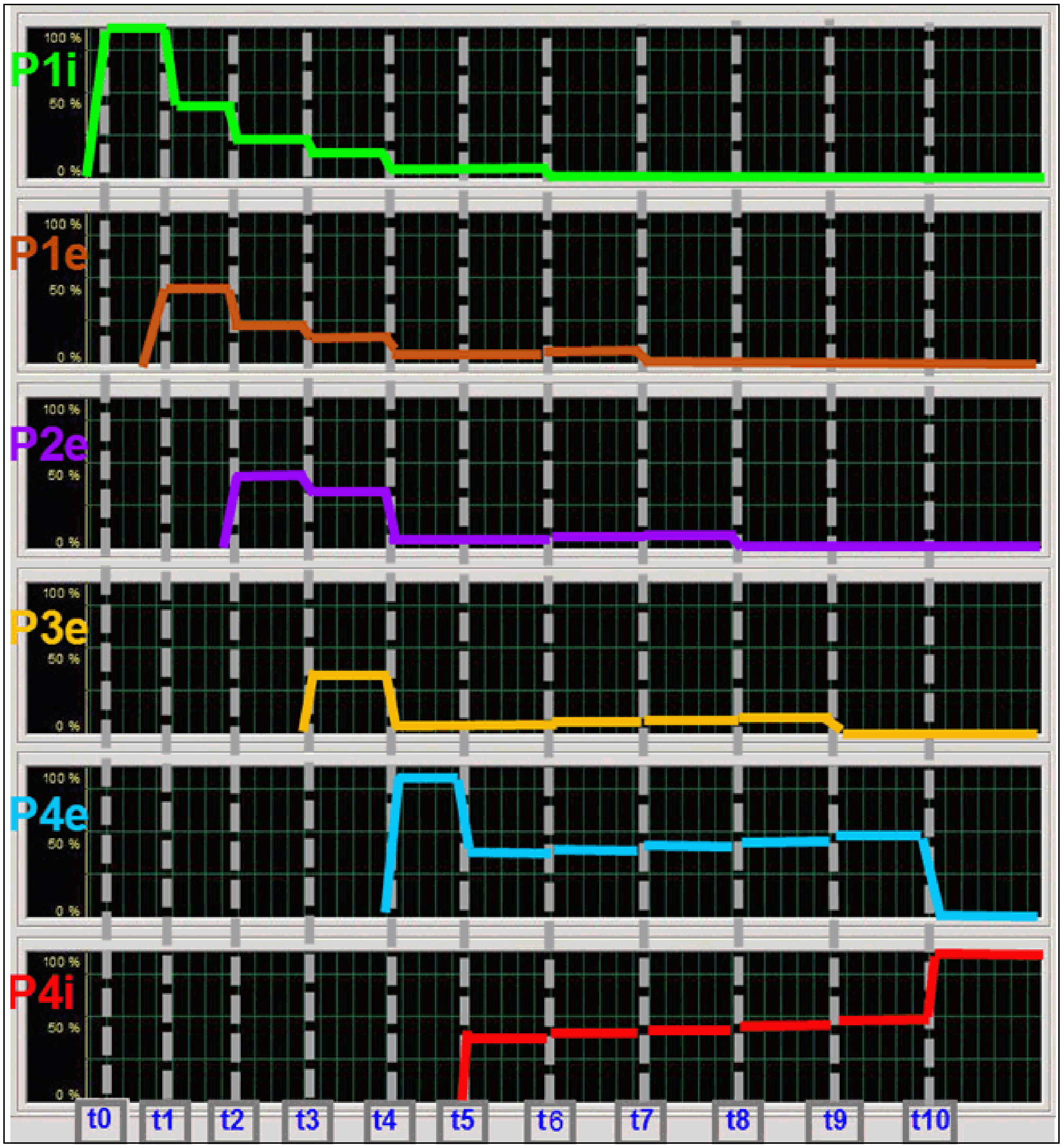
Table 9: Non-DCB Oversubscription With One High Priority Partition

Port, Partition	Relative Bandwidth Weight (%)	Maximum Bandwidth	Protocol	Plot Color
Port 0, Partition 1 (P1i)	1	100	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	1	100	Ethernet	Brown
Port 0, Partition 2 (P2e)	1	100	Ethernet	Purple
Port 0, Partition 3 (P3e)	1	100	Ethernet	Yellow
Port 0, Partition 4 (P4e)	97	100	Ethernet with TOE	Blue
Port 0, Partition 4 (P4i)	97	100	iSCSI Offload	Red

The following plot shows a similar effect to some of the previous examples, except that the fourth partition takes as much of the bandwidth as it needs (up to ~100%) when it starts to transmit. In this example the three 0% Relative Bandwidth Weight partitions have an effective Relative Bandwidth Weight of 1% instead of 0%.

- The first partition's traffic flow (P1i) initially takes ~100% of the available bandwidth when the test application starts to transmit traffic on that port by itself at **t0**.
- When P1e starts to send Ethernet traffic at **t1**, both will stabilize to ~5 Gbps each.
- When partition P2 (traffic flow P2e) starts to send traffic at **t2**, since only partition's P1 and P2 are now sending each partition gets half of the send bandwidth, so partition P2 (P2e) gets all of the allocated ~5 Gbps and partition P1's two traffic flows (P1i and P1e) will share it's allocated ~5 Gbps or ~2.5 Gbps each.
- When partition P3 (P3e) starts to send at **t3**, all three partitions will be allocated 1/3rd of the available bandwidth - P3e will received ~3.3 Gbps, P2e will receive the same allocation of ~3.3 Gbps and P1i an P1e will approximately split it's partition's bandwidth for ~1.65 Gbps each.
- But when P4e starts to send Ethernet traffic at **t4**, it will take almost all of the ~10 Gbps bandwidth, regardless of the bandwidth needs of the other three partitions four traffic flows. P1i and P1e will each get approximately half of 1/97 of the available bandwidth or ~0.05 Gbps while P2e and P3e will receive ~0.1 Gbps and P4e will take ~9.7 Gbps.
- When P4i starts to send Ethernet traffic at **t5**, it will take half of the allocated bandwidth for partition P4. Therefore P4e will drop to ~4.75 Gbps and P4i will start at ~4.75 Gbps. The other three partitions four traffic flows will be unaffected. P1i and P1e will each get approximately half of 1/103 of the available bandwidth or ~0.05 Gbps while P2e and P3e will receive ~0.1 Gbps and P4e will take ~9.7 Gbps.
- When P1i stops sending traffic at **t6**, it's freed up 0.05 Gbps bandwidth will be reallocated to the other traffic flows according to their relative bandwidth weight settings. The same is true for when P1e (at **t7**), P2e (at **t8**) and P3e (at **t9**) stop sending traffic.

- Finally when P4e stops sending traffic at **t10**, P4i will take all of the available bandwidth for ~10 Gbps.
- Whenever the fourth partition's bandwidth needs drop off, the other actively sending partitions will equally increase their respective shares to automatically occupy all of the available bandwidth.



Default Fixed Subscription Example

This is an example of the default partition settings that has all of the relative bandwidth weights set to 0% and the maximum bandwidths set to 2.5 Gbps. Since the total of the maximum bandwidth values are set to exactly 100% (i.e. never can reach an oversubscription situation), the traffic flows in each partition will share that partition's bandwidth allocation with respect to it's overall maximum bandwidth ceiling and the relative bandwidth weights are never used.

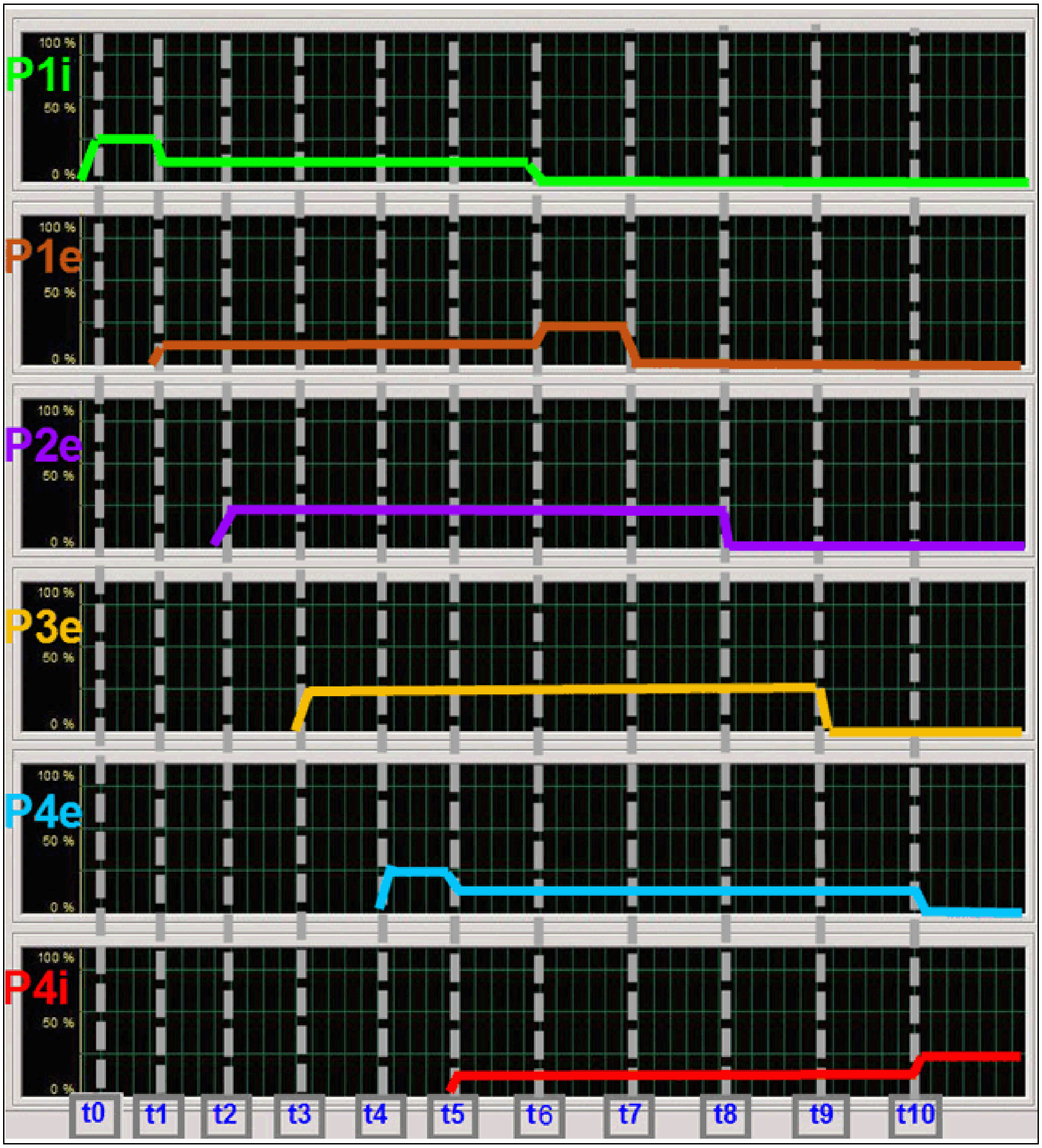
Table 10: Non-DCB Default Fixed Subscription

Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	0	25	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	0	25	Ethernet	Brown
Port 0, Partition 2 (P2e)	0	25	Ethernet	Purple
Port 0, Partition 3 (P3e)	0	25	Ethernet	Yellow
Port 0, Partition 4 (P4e)	0	25	Ethernet	Blue
Port 0, Partition 4 (P4i)	0	25	iSCSI Offload	Red

This following plot shows how the four partition's send traffic is independent of each other. Unlike the previous examples, none of the partitions (and their associated traffic flows) take more than their designated bandwidth portion; the total bandwidth of all four partitions of the port is equal to or less than the total available bandwidth of the port. In this example, each partition takes only ~25% or 2.5 Gbps of the total available bandwidth when their test application starts to transmit traffic. Furthermore, if a partition has more than one active traffic flow, these flows will share that partition's allowed bandwidth. Unused port bandwidth is not reallocated to any partition above it's own maximum bandwidth setting.

- When P1i starts to send traffic at **t0**, it only takes the subscribed 25% of the 10 Gbps bandwidth available which is ~2.5 Gbps.
- when P1e starts to send at **t1**, it will share partition P1's 25% with P1i. Each is allocated ~1.25 Gbps and neither expands into the unused ~7.5 Gbps remaining.
- When P2e starts to send at **t2**, it only takes it's partitions subscribed ~2.5 Gbps and does not affect either of partition P1's sending traffic flows.
- When P3e starts to send at **t3**, it again only takes it's partitions subscribed ~2.5 Gbps and does not affect P2e or either of partition P1's sending traffic flows.
- When P4e starts to send at **t4**, it also only takes it's partitions subscribed ~2.5 Gbps and does not affect any of the other partition's sending traffic flows.
- When P4i starts to send at **t5**, it will share partition P4's 25% with P4e. Each is allocated ~1.25 Gbps and the other partitions are unaffected.
- When P1i stops sending at **t6**, it will release it's 12.5% share of the bandwidth and the other remaining partition P1 traffic flow (P1e) will increase to 2.5 Gbps while the other traffic flows are unaffected.
- When P1e stops sending at **t7**, there will only be three partitions, but each is still assigned only 25% of the overall bandwidth. The other traffic flows (P2e, P3e, P4e and P4i) will not change.
- When P2e stops sending at **t8**, again there will be no change to the other traffic flows.
- When P3e stops sending at **t9**, there will still be no change to the other traffic flows.

- When P4e stops sending at **t10**, the remaining traffic flow on P4 (P4i) will absorb the freed 12.5% of the partition P4's allocated bandwidth and will increase to 2.5 Gbps.
- Each partition's flows on the same port are logically isolated from the others as if they were on separate ports and a partition's send flows stopping or restarting will not affect its fellow partition's send traffic flows - except where the flows are on the same partition - and then they will only take the freed bandwidth for their respective partition.



Mixed Fixed Subscription and Oversubscription Example

This example shows partitions with all of the relative bandwidth weights set the same, but with the partitions partially oversubscribing the available bandwidth, unequally. Two of the partitions are set to use 10% or 1 Gbps each of bandwidth and the other two (the ones with the hardware offload protocol's enabled) are set to use 80% or 8 Gbps of the connection's bandwidth, thus oversubscribing the connection by 80%.

Table 11: Non-DCB Mixed Fixed Subscription and Oversubscription

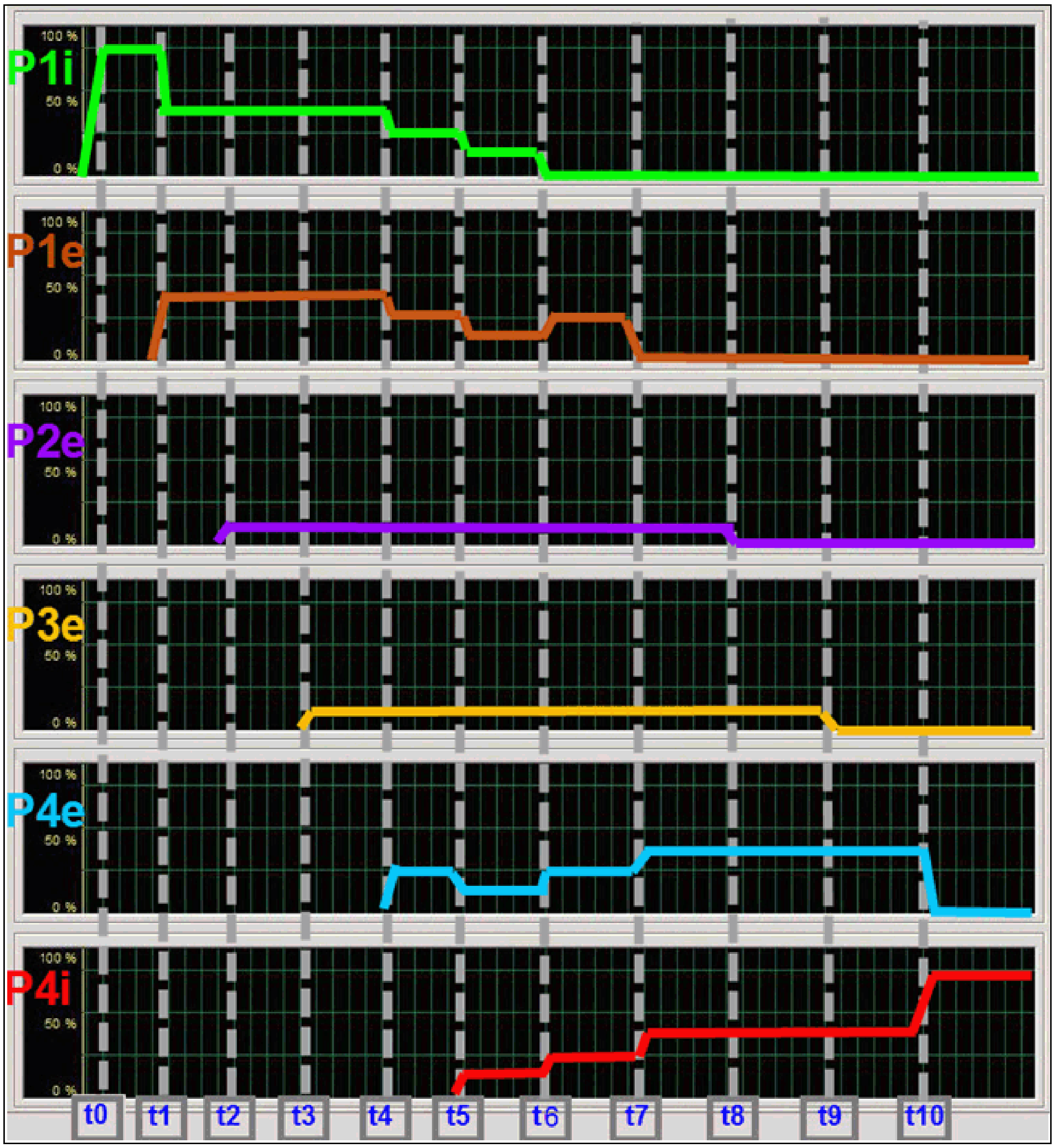
Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	0	80	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	0	80	Ethernet	Orange
Port 0, Partition 2 (P2e)	0	10	Ethernet	Purple
Port 0, Partition 3 (P3e)	0	10	Ethernet	Yellow
Port 0, Partition 4 (P4e)	0	80	Ethernet	Blue
Port 0, Partition 4 (P4i)	0	80	iSCSI Offload	Red

This is a combination example of a fixed subscription (three of the partitions sum to 100%), but all four sum to 180%. When all four, or at least the two larger partitions, are running traffic, they share the space with each other, up to their partition maximum bandwidth values; otherwise, they act as if they are independent connections.

- The first partition's traffic flow (P1i) initially takes its designated ~8 Gbps when the test application starts to transmit traffic at **t0** to that port by itself, not expanding into the remaining unused ~2 Gbps bandwidth.
- When the second traffic flow on the first partition (P1e) starts to send at **t1**, the two active traffic flows on the same partition share its ~8 Gbps bandwidth for ~4 Gbps each.
- When the third traffic flow (P2e) starts sending at **t2**, it only takes its partitions maximum bandwidth allowed ~1 Gbps. Partition P1's two traffic flows are unaffected.
- When the fourth traffic flow (P3e) starts sending at **t3**, it again only takes its partitions maximum bandwidth allowed ~1 Gbps. Partition P1's two traffic flows and the traffic flow on partition P2 (P2e) are unaffected.
- But when P4e starts to send traffic at **t4**, the condition is now oversubscribed. Since P2e and P3e uses only ~1 Gbps of their allocated 2 Gbps (10 Gbps / 5 equally weighted traffic flows) that leaves ~8 Gbps free for the other three traffic flows. The remaining traffic flows (P1i, P1e and P4e) would then be allocated ~2.6 Gbps each (8 Gbps / 3 equally weighted traffic flows).
- But when P4i starts to send traffic at **t5**, it shares the available bandwidth within it's maximums with the other traffic flows. P2e and P3e are still using only ~1 Gbps of their allocated 1.6 Gbps (10 Gbps / 6 equally weighted traffic flows) which again leaves ~8 Gbps free for the other four traffic flows. Therefore these four (P1i, P1e, P4e and P4i) are allocated ~2 Gbps each (8 Gbps / 4 equally weighted traffic flows).
- When P1i stops at **t6**, it releases it's bandwidth to the available pool and since P2e and P3e are capped by their maximum bandwidth value to 1 Gbps, the three other traffic flows (P1e, P4e and P4i) automatically take equal shares and increase their bandwidth used to ~2.6 Gbps each.
- When P1e subsequently stops sending at **t7**, P4e and P4i grab up the extra available bandwidth and go to ~4 Gbps each. Both P2e and P3e are unaffected and continue sending at ~1 Gbps each.
- When P2e stops sending at **t8**, P4e and P4i are not able to make use of the freed up bandwidth since they

are both in partition P4 which as a maximum bandwidth ceiling of 8 Gbps. Therefore none of the traffic flows increase their sending rates and this unused bandwidth is ignored.

- When P3e stops sending at **t9**, the same condition is still in effect. Therefore none of the remaining active traffic flows increase their sending rates to use this extra bandwidth.
- Finally, P4e stops at **t10** and this allows it's companion traffic flow (P4i) to increase to ~8 Gbps which is partition P4's maximum top end. The remaining ~2 Gbps is unassigned.



The following example is the same as the previous example, but with FCoE in the first partition. Additionally, the FCoE traffic flow is Lossless and in DCB Priority Group 1 with an ETS = 50% and the other traffic flows are Lossy and in Priority Group 0 with an ETS = 50%.

Table 12: DCB Mixed Fixed Subscription and Oversubscription with Lossless FCoE Offload

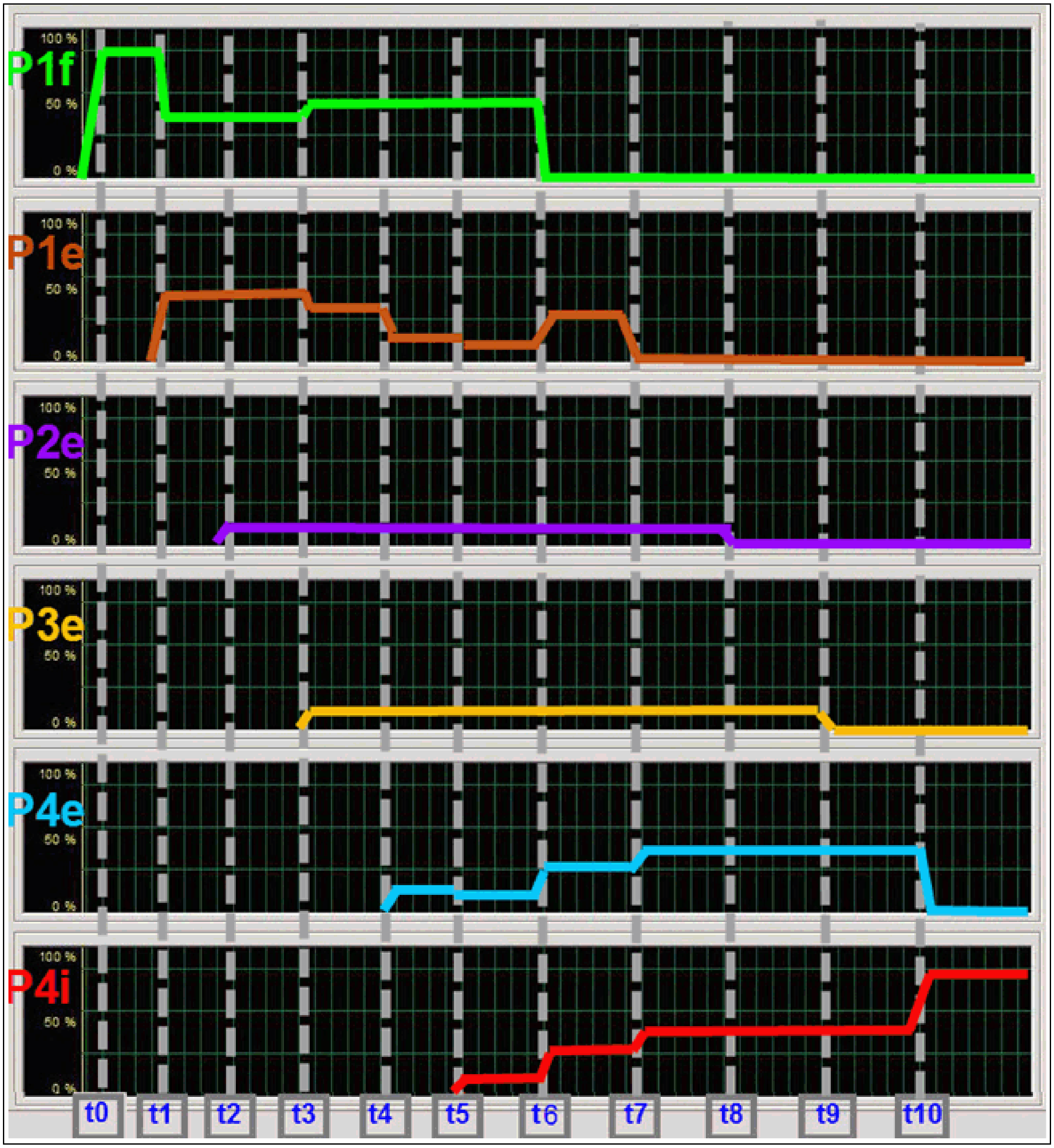
Port, Partition	Relative Bandwidth Weight (%)	Maximum Bandwidth	Protocol	Plot Color
Port 0, Partition 1 (P1f)	N/A	80	FCoE Offload	Green
Port 0, Partition 1 (P1e)	N/A	80	Ethernet	Orange
Port 0, Partition 2 (P2e)	N/A	10	Ethernet	Purple
Port 0, Partition 3 (P3e)	N/A	10	Ethernet	Yellow
Port 0, Partition 4 (P4e)	N/A	80	Ethernet	Blue
Port 0, Partition 4 (P4i)	N/A	80	iSCSI Offload	Red

This is a similar combination example of a fixed subscription (three of the partitions sum to 100%), but all four sum to 180%. When all four, or at least the last two partitions, are running traffic, they share the space with each other, up to their partition maximum bandwidth values and their PG's ETS settings; otherwise, they act as if they are independent connections.

- The first partition's traffic flow (P1f) initially takes its maximum bandwidth designated ~8 Gbps when the test application starts to transmit traffic at t0 to that port by itself, not expanding into the remaining unused ~2 Gbps bandwidth.
- When the second traffic flow on the first partition (P1e) starts to send at **t1**, the two active traffic flows on the same partition share its ~8 Gbps bandwidth for ~4 Gbps each. ETS does not take effect since the traffic in PG0 and PG1 are still less than the amount prescribed by their respective ETS values.
- When the third traffic flow (P2e) starts sending at **t2**, it only takes its partitions maximum bandwidth allowed which is ~1 Gbps. Partition P1's two traffic flows are unaffected and the unassigned 1 Gbps bandwidth remains free.
- When the fourth traffic flow (P3e) starts sending at **t3**, it only takes its partitions maximum bandwidth allowed which is ~1 Gbps. Now the first partition's two traffic flows readjust so that PG 0 does not get more than 50% of the overall bandwidth - i.e. PG0's P1e+P2e+P3e = 40%+10%+10% which is greater than 50%. The P1e traffic flow is reduced to 30% or ~3 Gbps and the P1f traffic flow (in PG1) is adjusted upwards to 50% or ~5 Gbps.
- When P4e starts to send traffic at **t4**, it equally shares PG0's ETS assigned bandwidth with P1e, P2e and P3e but since P2e and P3e use only ~1 Gbps of their allocated 1.25 Gbps (5 Gbps / 4 equally weighted traffic flows) this leaves ~3 Gbps free (5 Gbps available - 2 GBps assigned to P2e and P3e) for the other two traffic flows (P1e and P4e) and they are both allocated ~1.5 Gbps each (3 Gbps / 2 equally weighted traffic flows). P1f is in PG1 so it is unaffected and keeps sending at ~5 Gbps.
- When P4i starts to send traffic at **t5**, it also equally shares PG0's bandwidth (5 Gbps / 5 equally weighted traffic flows) which means P1e, P2e, P3e, P4e and P4i all send at ~1 Gbps. P1f in PG1 is still unaffected and keeps sending at ~5 Gbps.
- When P1f stops at **t6**, it releases all of PG1's bandwidth to the available pool and since P2e and P3e are capped by their maximum bandwidth value to 1 Gbps, the three other traffic flows (P1e, P4e and P4i) automatically take equal shares of 8 Gbps and bump up their bandwidth used to ~2.6 Gbps each.
- When P1e subsequently stops sending at **t7**, P4e and P4i grab up the extra available bandwidth and go to ~4 Gbps each. Both P2e and P3e are unaffected and continue sending at ~1 Gbps each.
- When P2e stops sending at **t8**, P4e and P4i are not able to make use of the freed up bandwidth since they are both in partition P4 which has a maximum bandwidth ceiling of 8 Gbps. Therefore none of the traffic

flows increase their sending rates and the unused bandwidth is ignored.

- When P3e stops sending at **t9**, the same condition is still in effect. Therefore none of the remaining active traffic flows increase their sending rates to use this extra bandwidth.
- Finally, P4e stops at **t10** and this allows it's companion traffic flow (P4i) to increase to ~8 Gbps which is partition P4's maximum top end. The remaining ~2 Gbps is unassigned.



Mixed Weights and Subscriptions Example

This example shows partitions with different relative bandwidth weights and maximum bandwidths, but with the same partitions partially oversubscribing of the available bandwidth as the previous example. The first pair of partitions are set to use 10% or 1 Gbps each of bandwidth and both of their weights are set to 5% while the second pair of partitions are set to use 80% or 8 Gbps of the connection's bandwidth each with both of their relative bandwidth weights set to 45%. The total is still oversubscribing the connection by 80%.

In DCB mode, there would be no difference between the previous DCB mode example and this one, since the Relative Bandwidth Weights are not applicable (in DCB mode) and also if all of the traffic types are in the same Priority Group; the results would be similar.

Table 13: Non-DCB Mixed Fixed Subscription and Oversubscription

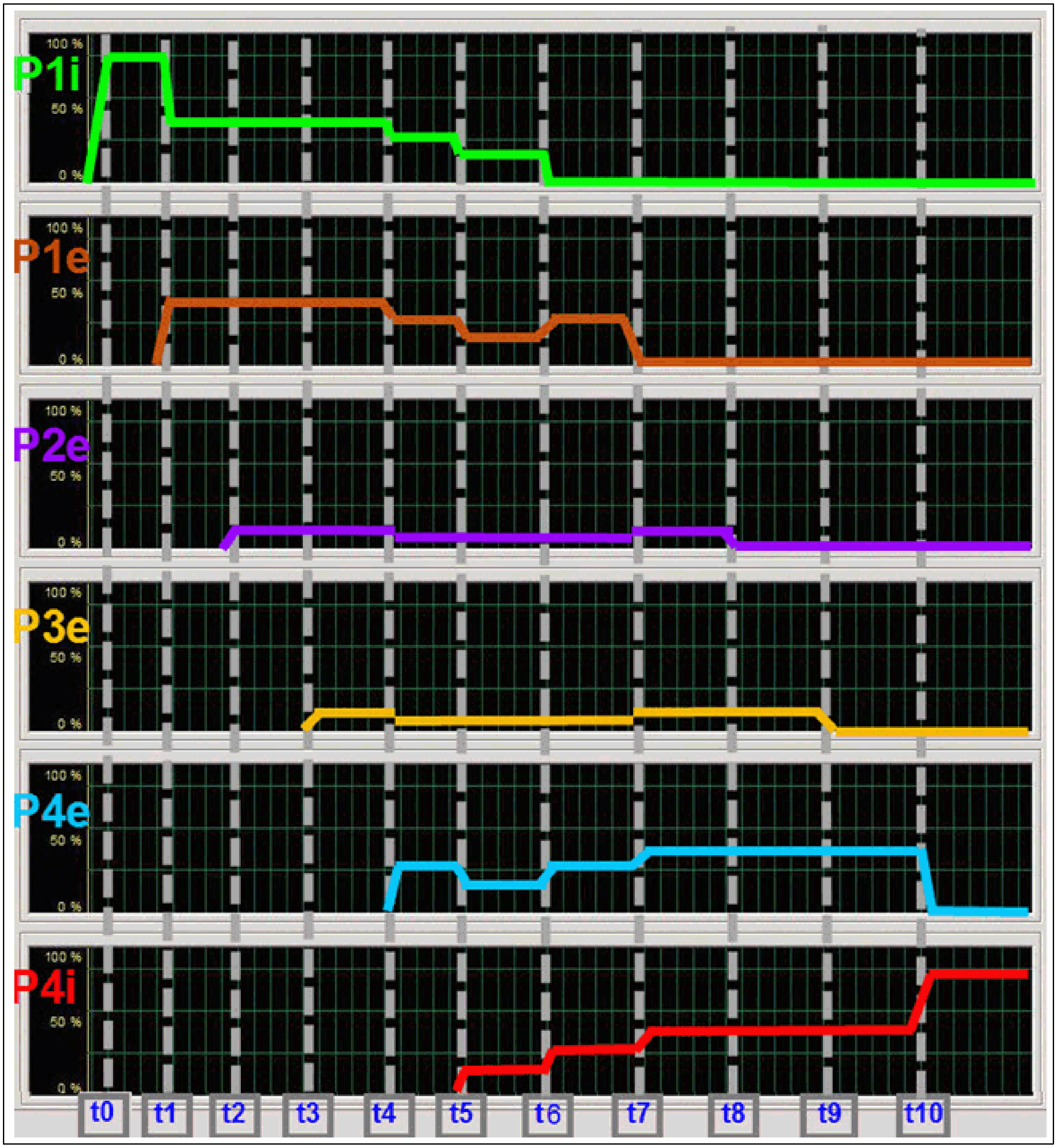
Port, Partition	*Relative Bandwidth Weight (%)*	*Maximum Bandwidth*	*Protocol*	*Plot Color*
Port 0, Partition 1 (P1i)	45	80	iSCSI Offload	Green
Port 0, Partition 1 (P1e)	45	80	Ethernet	Orange
Port 0, Partition 2 (P2e)	5	10	Ethernet	Purple
Port 0, Partition 3 (P3e)	5	10	Ethernet	Yellow
Port 0, Partition 4 (P4e)	45	80	Ethernet	Blue
Port 0, Partition 4 (P4i)	45	80	iSCSI Offload	Red

This is a combination example of a fixed subscription (three of the partitions sum to 100%), and oversubscription (all four sum to 180%) with different weights and maximum bandwidths. When all four, or at least the two larger partitions, are running traffic, they share the space with each other with respect to their partition's weight and maximum bandwidth values; otherwise, the partition's continue to act as if they are independent connections.

- The first partition's traffic flow (P1i) initially takes its designated ~8 Gbps when the test application starts to transmit traffic at **t0** to that port by itself, not expanding into the remaining unused ~2 Gbps bandwidth.
- When the second traffic flow on the first partition (P1e) starts to send at **t1**, the two active traffic flows on the same partition share its ~8 Gbps bandwidth for ~4 Gbps each.
- When the third traffic flow (P2e) starts sending at **t2**, it only takes its partitions maximum bandwidth allowed ~1 Gbps. Partition P1's two traffic flows are unaffected.
- When the fourth traffic flow (P3e) starts sending at **t3**, it again only takes its partitions maximum bandwidth allowed ~1 Gbps. Partition P1's two traffic flows (P1i and P1e) and the traffic flow on partition P2 (P2e) are unaffected.
- But when P4e starts to send traffic at **t4**, the traffic needs are oversubscribed so the available bandwidth is redistributed based on each partition's individual weights and maximums settings. P2e and P3e use 5% each (5/100) so their traffic flows are reduced to ~0.5 Gbps which leaves ~9 Gbps free for the other three traffic flows. The two other partition's traffic flows are allocated ~3 Gbps each (9 Gbps / 3 equally weighted traffic flows) - the total bandwidth for P1i and P1e is 6 Gbps which is less than partition P1's maximum of 80% of 10 Gbps.
- When P4i starts to send traffic at **t5**, the bandwidth is again redistributed. P2e and P3e are still using only ~0.5 Gbps (5/100). This again leaves ~9 Gbps free for the remaining four equally weighted traffic flows, therefore these four (P1i, P1e, P4e and P4i) all are allocated ~2.25 Gbps each (9 Gbps / 4 flows) where P1i

plus P1e and P4e plus P4i totals are 4.5 Gbps each which is less than their respective partition's maximum bandwidth settings.

- When P1i stops at **t6**, it releases it's bandwidth to the available pool and since P2e and P3e are capped by their relative bandwidth weight values to 0.5 Gbps, the three other traffic flows (P1e, P4e and P4i) automatically take equal shares of the remaining bandwidth and bump up their portion to ~3 Gbps each where P4e plus P4i total is 6 Gbps which is still less than their respective partition's maximum bandwidth value.
- When P1e subsequently stops sending at **t7**, P4e and P4i grab up some of the extra available bandwidth and go to ~4 Gbps each, where they reach their partition's maximum bandwidth value of 80% or 8 Gbps. The remaining bandwidth is shared equally by P2e and P3e at ~1 Gbps each.
- When P2e stops sending at **t8**, P4e and P4i are not able to make use of the freed up bandwidth since they are both in partition P4 which as a maximum bandwidth ceiling of 8 Gbps. The same is true for P3e which is also at it's bandwidth maximum. Therefore none of the remaining traffic flows increase their sending rates and this unused bandwidth is ignored.
- When P3e stops sending at **t9**, the same maximum ceiling condition is still in effect. Therefore neither P4e or P4i increase their sending rates to use this extra bandwidth.
- Finally, P4e stops at **t10** and this allows it's companion traffic flow (P4i) to increase to ~8 Gbps which is partition P4's maximum top end. The remaining ~2 Gbps is unassigned.


P1i
P1e
P2e
P3e
P4e
P4i
100 %
50 %
0 %
t0
t1
t2
t3
t4
t5
t6
t7
t8
t9
t10

–

Dell has tested and certified the Broadcom 57712-k dual-port 10 GbE KR Blade Converged Network Daughter Card, Broadcom 57810S dual-port 10GbE SFP+ Converged Network Adapter, Broadcom 57810S-k dual-port 10GbE KR Blade Converged mezzanine card, Broadcom 57810S-k dual-port 10GbE KR Blade Converged Network Daughter card, and Broadcom 57800S quad-port SFP+ / RJ45 (2x10GbE + 2x1GbE) Rack Converged Network Daughter Card, 57810S-T dual-port 10GBASE-T Converged Network Adapter, and 57800S-T quad-port 10GBASE-T / RJ45 (2x10GbE + 2x1GbE) Rack Converged Network Daughter Card with Switch Independent Partitioning, TOE, CEE/DCB, iSCSI Offload (iSCSI Offload-TLV) and FCoE Offload technology. Dell specifically disclaims knowledge of the accuracy, completeness, or substantiation for all statements and claims made in this document regarding the properties, speeds, or qualifications of the adapter.

Broadcom® Corporation reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design.

Connecting everything®

BROADCOM CORPORATION
5300 California Avenue
Irvine, CA 92617

Phone: 949-926-5000
Fax: 949-926-5203
E-mail: info@broadcom.com
Web: www.broadcom.com

2CS57712-SWUM103-R June 07, 2012